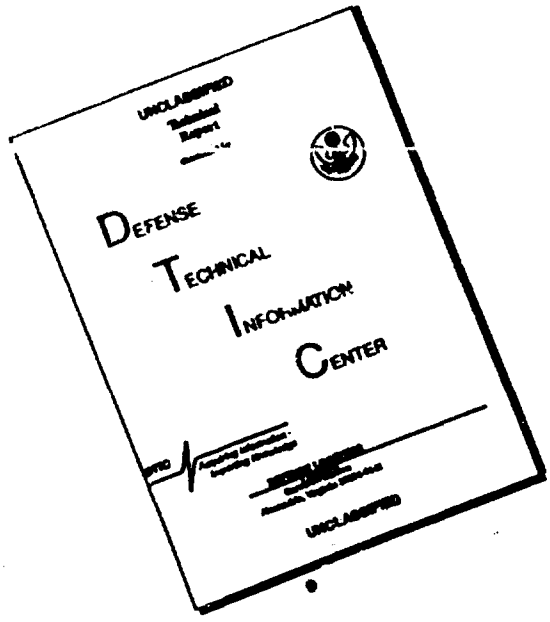
Finance User Interface Style Guide (3270) to the Defense Information Technology Services Organization

Version 1.0
November 1992

Coordination Draft

DISCLAIMER NOTICE



THIS DOCUMENT IS BEST QUALITY AVAILABLE. THE COPY FURNISHED TO DTIC CONTAINED A SIGNIFICANT NUMBER OF PAGES WHICH DO NOT REPRODUCE LEGIBLY.

REPORT DOCUMENTATION PAGE

Form Approved
OMB No. 0704-0188

1. AGENCY USE ONLY (Leave blank)	2. REPORT DATE	3. REPORT TYPE AND DATES COVERED
	November 1992	final

4. TITLE AND SUBTITLE
Finance User Interface Style Guide (3270) to the Defense Information Technology Services Organization

5. FUNDING NUMBERS

6. AUTHOR(S)
Jerry Bennis

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)
Defense Information Systems Agency (DISA)
Center for Information Management (CIM)
Office of Technical Integration (OTI)

8. PERFORMING ORGANIZATION REPORT NUMBER

9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)
DISA/CIM/OTI
5201 Leesburg Pike, Suite 1501
Falls Church, VA 22041-3201

10. SPONSORING/MONITORING AGENCY REPORT NUMBER

11. SUPPLEMENTARY NOTES

12a. DISTRIBUTION/AVAILABILITY STATEMENT
Available to the public

12b. DISTRIBUTION CODE

13. ABSTRACT *(Maximum 200 words)*

This document presents to the Finance migration technical community guidelines on how to best develop or reengineer user interfaces for Finance applications. During consolidation of Finance legacy systems, every attempt should be made to have the applications provide a standardized interface (e.g., a common "look and feel") to the end users. This will reduce training requirements as users move from one Finance application to another. While a majority of selected Finance systems are IBM 3270-based, they were developed using different user interface guidelines. To ensure a standard approach for application development, or reengineering, during the consolidation, a "Finance User Interface Style Guide (3270)" has been developed.

14. SUBJECT TERMS
Technical Integration
User Interface
Finance
(CIM) Collection

15. NUMBER OF PAGES
166

16. PRICE CODE

17. SECURITY CLASSIFICATION OF REPORT	18. SECURITY CLASSIFICATION OF THIS PAGE	19. SECURITY CLASSIFICATION OF ABSTRACT	20. LIMITATION OF ABSTRACT
unclassified	unclassified	unclassified	

Defense Information Technology Services Organization (DITSO)
6760 East Irvington Place
Denver, Colorado 80279-1000

Provided By: **Defense Information Systems Agency**
Center for Information Management
Office of Technical Integration
5201 Leesburg Pike, Suite 1501
Falls Church, Virginia 22041

November 1992

Task Order: 3-OTI-2, 9-002-C

TABLE OF CONTENTS

Section 1

Section 2

Section 3

Chapter 4

Chapte 19

Section 1
Finance Migration
User Interface Guidance

1.1 Background

The purpose of this paper is to present the Finance Migration technical community with guidelines on how to best develop or reengineer user interfaces for Financial applications.

During the consolidation of Finance legacy systems, every attempt should be made to have the applications provide a standardized interface (e.g., a common "look and feel") to the end users. This will reduce training requirements as users move from one Finance application to another. While a majority of selected Finance systems are IBM 3270-based, they were developed using different user interface guidelines. To ensure a standard approach for application development, or reengineering, during the consolidation, a "Finance User Interface Style Guide (3270)" has been developed.

This document was developed after reviewing existing Finance user interface guidance, such as: "User Interface Design Standards for the Defense Travel Pay System," APCAPS Guideline STD006003 "Human Computer Interface," APCAPS Guideline STD013001 "Message Handling," DDMS screen guidelines (Task Codes BC7, BC11, etc.), DTRS Screen Design guidelines (A25), DJMS sample screens, and "DRAS On-line Standard Edits." Based on common practices found in these guidelines, the "Finance User Interface Style Guide (3270)" was developed to provide guidance for 3270 text-based applications, 3270 emulation, and applications treated as extensions to 3270 applications (e.g., data is downloaded from mainframe to a PC, PC application then uses the data). This 3270 guidance is intended for use only when a Finance application is being developed or an existing application is being redeveloped; it is not the intention of this guidance that existing applications be changed just to comply with this guidance.

The "Finance User Interface Style Guide (3270)" covers a number of optional techniques available in the 3270 environment. This includes action bars, pull-down windows, pop-up windows, and use of a mouse. While these techniques were not used in many 3270 applications, it is the intent of guidance to cover these with anticipation that new 3270 development, or redevelopment, might make use of these techniques. Use of these techniques can be viewed as a step toward a more user-friendly interface. The use of these techniques is not required, but its use incurs no additional costs or time. They are simply different ways of designing/developing a user interface. Use of these techniques can be part of a migration strategy of user interfaces.

1.2 Migration Strategy

The future should include the evolution of applications that have a single look and feel; a single log-on to the system, multi-media capabilities; human portability; and the ability to log on from any location and become productive. These and related topics are covered in the DoD Human Computer Interface Style Guide and in the DoD Technical Architecture Framework for Information Management Volume 2. The "Finance User Interface Style Guide (3270)" deals with the short-term consolidation, with an eye toward the previously mentioned "DoD Style Guide" and "Framework." Other measures must be taken during the near-term consolidation of Finance systems to provide the optimum, longer term user interface.

Investments made in new development within the Finance area should be aimed at open system approaches, to ensure the best return on the investment. While at times it may not be technically feasible to deliver an open systems user interface (MOTIF, X Windowing, etc.), alternatives are possible.

It is possible to deliver a user interface compliant with the DoD Human Computer Interface Style Guide (MOTIF) with 3270-type applications. "False-front" software (COTS) provides the designer/developer a means of delivering a graphical user interface (GUI), such as MOTIF, on PCs. The false-front software captures 3270 data streams and offers tools to the developer for building GUIs on the 3270 streams. False-fronts provide a compromise from the current use of proprietary 3270 applications, and the ambitious goal of building X Windowing system applications. False-front development requires no changes to current mainframe applications; development of the GUI is on the PC only.

Guidance for use of false-fronts is covered in the "Finance Migration, False-Front Guidance."

1.3 Why Move to a Graphical User Interface?

In 1986, Peat Marwick & Co. conducted a study to evaluate productivity and user benefits of character-based applications (DOS) and graphical-based applications (Macintosh). The conclusions were that:

- The time and effort to learn additional applications were reduced for the graphical-based applications
- User acceptance, especially among professional workers, was higher for the graphical-based applications
- The graphical-based interface provided a means to integrate applications

A second study in 1988 by Solutions, Inc. concluded that a graphical user interface accounted for savings in training and support. Other industry studies have provided strong evidence that a GUI can improve productivity by as much as 40 percent over a character-based environment (Chapter 9 of DMR's "Stage One, Open Systems Status Report").

A 1990 User Interface Engineering study commissioned by Microsoft and Groupe Bull found that both novice and experienced microcomputer users worked faster and made fewer mistakes with a GUI than those only equipped with a text-based user interface. Experienced users were found to be 58 percent more productive and novice users 48 percent more productive (CIO - July 1992).

The "Policy for Graphical User Interfaces (GUI)," 24 February 1992, from the Director of Defense Information, states that, "Beginning immediately, planning for all DoD systems will evaluate application of Federal Information Processing Standard (FIPS) 158 (X Windows) . . ."

Section 2
"False-Front" Guidance

2.1 Background

When it has been decided that an application will use a graphical user interface (GUI), there are at least two approaches to be considered: 1. migrate to a client/server architecture, using windowing clients, 2. or use "false-front" software. For most shops, the first choice includes redevelopment, integration, training, data migration and much time. (Refer to the DoD Human Computer Interface Style Guide for this choice.) The second choice leaves the current mainframe application untouched, but delivers a graphical user interface to the user. The false-front approach does not make use of a client/server architecture or SQL data access, but instead intercepts and interprets host application screens, remapping them as graphical windows.

False-front technology is particularly useful for projects attempting to migrate from existing text-based applications to a graphical user interface environment.

2.2 Use of False-Front Technology

A false-front tool is used to "paint" the screens. This is done, with some of the tools, by letting the tool capture some 3270-based screens from the host applications and then remake them with radio buttons, icons, and other GUI options. When finished, the false-front application runs on the PC, communicating with the host application via a terminal emulation board or asynchronous link. The result is a common "look and feel" graphical user interface for the user across applications, without touching the host applications.

Applications using a false-front tool can at a future date be transitioned to a client/server environment; the user interface piece (typically part of a client) has already been developed. With some of the tools, it is also possible to integrate with PC-based applications (e.g., Lotus 1-2-3, etc.).

A list of currently available false-front tools is on the next page. It is recommended that a tool supporting Motif be used, but in some circumstances it may be practical, for short-term benefits, to use a tool that supports Microsoft Windows.

2.3 Benefits of Using False-Front Technology

There are short-term benefits that can be realized from the use of these tools:

- Keymapping problems can sometimes be eased because the key, radio buttons, etc., are located from the host application and can be configured on the PC.
- The user interface is isolated from the host application. Changes to screen layouts and data entry edits can be made at the PC-level instead of making changes to the host code.
- Designers and developers can gain valuable training and experience in graphical interface development, which will be useful in the open systems future.

A Selected List of GUI False-Front Builders			
Product/Vendor	Motif Support	MS Windows Support *	Price **
WindowsMaker Professional Blue Sky Software Corp.	no	yes	$995
Omnis 5 Blyth Software Inc.	no	yes	$795
Objectvision Borland International Inc.	no	yes	$150
CASE:W CASEworks Inc.	no	yes	$995
Easel Easel Corp.	no	yes	N/A
Guild for Windows Expert-EASE Systems Inc.	yes	yes	$1,995
Choreographer GUIdance Technologies Inc.	no	yes	$7,500
WinTran Guideware Corp.	yes	yes	$445
SQLWindows Gupta Technologies Inc.	no	yes	$1,295

A Selected List of GUI False-Front Builders			
Product/Vendor	Motif Support	MS Windows Support *	Price **
ObjectScript ObjectScript Professional ObjectView for Windows Matesys Corp.	no no no	yes yes yes	$150 $499 $899
Mozart Mozart Systems Corp.	no	yes	$7,000
Open Interface Neuron Data	yes	yes	$7,500
Aspect Open Inc.	yes	yes	$795
PowerBuilder Powersoft Corp.	no	no	$1,995
WinServe Rational Systems Inc.	yes	no	$5,000
WNDX The WNDX Corp.	no	yes	$295
Tigre Tigre Object Systems Inc.	yes	yes	$3,500
Flashpoint Viewpoint Systems Inc.	no	yes	$9,500
SQLFile Vizant Inc.	no	yes	$495
Rumba Tools for Visual Basic Wall Data Inc.	no	yes	$195
XVT XVT-Design for Windows XVT Software Inc.	yes yes	yes yes	$1,450 $1,200

* MS Support, supports Microsoft Windows 3.X

** The price listed varies for each product on what kinds of tools are included and number of PCs covered.

NOTE: See "Datamation" 1 July 1992

Section 3

Finance User Interface Style Guide (3270)

Chapter 1

Introduction

1.1 Background

This guideline defines the user interface to be used by application designers and developers for the IBM 3270 (text) environment, including: (1) 3270 emulation, (2) nonprogrammable terminals, and (3) programmable terminals used as extensions to mainframe applications. This guide details what is needed for people and computers to communicate with each other in a standard fashion, but not the details of how a program is written or generated to achieve this communication. The guidance largely explains what a user should be able to do or see. The way that communication is accomplished within the application is left to the designer or developer.

The IBM "Common User Access Panel Design and User Interaction" was used as the model for this document.

1.2 Benefits of a Consistent User Interface

This Style Guide is intended to provide a consistency across the Financial automated systems. This consistency has three parts:

1. **Physical consistency** - keyboard layout, the location of keys, use of mouse, etc. For example, the ***enter*** key would always be in the same place, regardless of the system being used.

2. **Syntactical consistency** - the sequence and order of appearance of elements on the screen and the sequence of action requests. For example, the current date would always appear in the upper right corner of a screen.

3. **Semantical consistency** - the elements that make up the interface. For example, ***return*** has the same meaning across applications (e.g., when a user selects ***return***, the same action occurs).

A consistent user interface saves time and money. Users benefit because they may need less time learning to use "new" Finance applications, if the interface is consistent across the applications. A consistent interface may reduce user frustrations and allow users to feel more comfortable with the system.

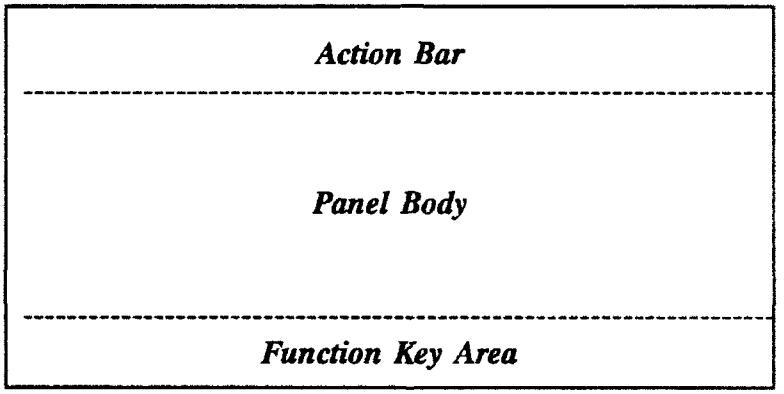
Application designers and developers benefit by defining common building blocks for an interface through standardized interface elements and interaction techniques. These building blocks can allow programmers to create and change application more easily and more quickly. Because the same panel can be used across many systems, application designers and developers can reuse panel designs and panel code.

Total consistency across different types of hardware is not always possible, so a standard user interface becomes a compromise between the desire for consistency and the capabilities available on a given type of hardware.

Chapter 2

Panel Design

A ***screen*** is the surface of the computer workstation or terminal on which information is displayed to a user. A ***panel*** is a predefined grouping of information arranged in a specified way and displayed on a screen. Different panel types are used to present different arrangements of information (menus, entry, information, list, and logo). Each panel has three areas: action bar/pull-downs, panel body, and function key area.



2.1 Panel Areas

The ***action bar*** appears at the top of a panel and provides a list of action choices. When a user selects one of the choices, a ***pull-down*** appears that lists available actions or functions. Pull-downs are extensions of the action bar. Action bars are optional.

Action bars and pull-downs can be useful to users by providing a visual cue instead of requiring a user to remember an action name. Selections from action bars and pull-downs can lead to a series of ***pop-up*** windows (sub-areas of the panel body) to request or provide additional information. There should be no more than three levels of pop-ups for any one action item. Pop-ups are optional.

The ***panel body*** is located below the action bar and above the function key area. Every panel will have a panel body. A panel body may be divided into several ***panel body areas*** to allow an application to show a user more than one group of information at a time, or to allow a user to enter or update more than one group of information at a time.

The panel body can also contain a ***command area***, in which a user types in application or system commands. The command area is an alternative way of giving command

interface (instead of using an action bar with pull-down). The ***message area*** provides feedback.

The ***function key area*** appears at the bottom of the panel and displays a list of function key assignments. A user may choose to display the function key area in short form or long form or not to display it at all. A function key area must be available for all panels.

Panel elements include panel titles, column headings, and other elements. These will be discussed later.

2.2 Panel Types

Menu Panel

Displays one or more lists of choices from which a user selects one or more choices. Contains selection fields for this purpose.

Entry Panel

Displays fields in which a user types information and selects choices. There are three styles:

> Parameter entry panel - displays fields for a user to type in parameters or specify options. Contains entry fields and selection fields for this purpose.
>
> Form fill-in panel - resembles paper forms and may contain several fields on each line.
>
> Tabular entry panel - displays fields arranged in rows and columns identified by headings.

Information Panel

Displays protected information such as data, messages, and help information.

List Panel

Displays list of items from which a user selects one or more choices and specifies one or more actions to work on these choices. Only one action at a time can work on different choices.

Logo Panel

Displays ownership and copyright information.

Panel Elements

Action bar

Lists choices that represent groups of actions a user can request. Actions appear in pull-downs. Used when two or more application actions are available.

Action bar pull-down

Extends action bar by displaying groups of actions a user may request. Appears when a user selects an action bar choice.

Check box

Is one of several indicators, with symbols next to selection field choices. When a user selects a choice, an "X" appears in the check box.

Entry field

Provides space in which users type information.

Field prompt

Identifies selection fields, entry fields, and variable data.

Function key area

Lists specified set of actions assigned to keys.

Headings

Identifies entry fields or selection fields. Column headings or group headings.

Instructions

Tell users what to do with data in panel body.

Message area

Displays messages in an area of the panel body.

Panel ID

Identifies user's position in the dialog.

Panel title

Identifies data in the panel body.

Pop-up window

Displays a panel, in portion of a screen, that extends the user's dialog with another window.

Scrolling information

Tells user that more information exists outside the visible panel body area and which direction to scroll.

Selected emphasis

Reminds users, with highlighting or symbols, what choices are selected.

Selection cursor

Highlights a selection field choice and entry field.

Selection field

Presents a group of related choices from which users make selections.

Text cursor

Indicates where typing input will occur in entry field.

Unavailable emphasis

Tells users what choices are unavailable for selection.

Chapter 3

Dialog Design

A dialog is a series of requests passed back and forth between the user and the computer. As the user and computer exchange requests, the dialog, under the users' control, moves along one of the paths provided by the application. Users "navigate" through the application using specific actions that are part of the dialog. These actions may request the computer to process information or simply cause movement through a session from one panel to another or from one application to another. Dialog actions also control what happens to information the user has typed in, and whether or not the information should be retained.

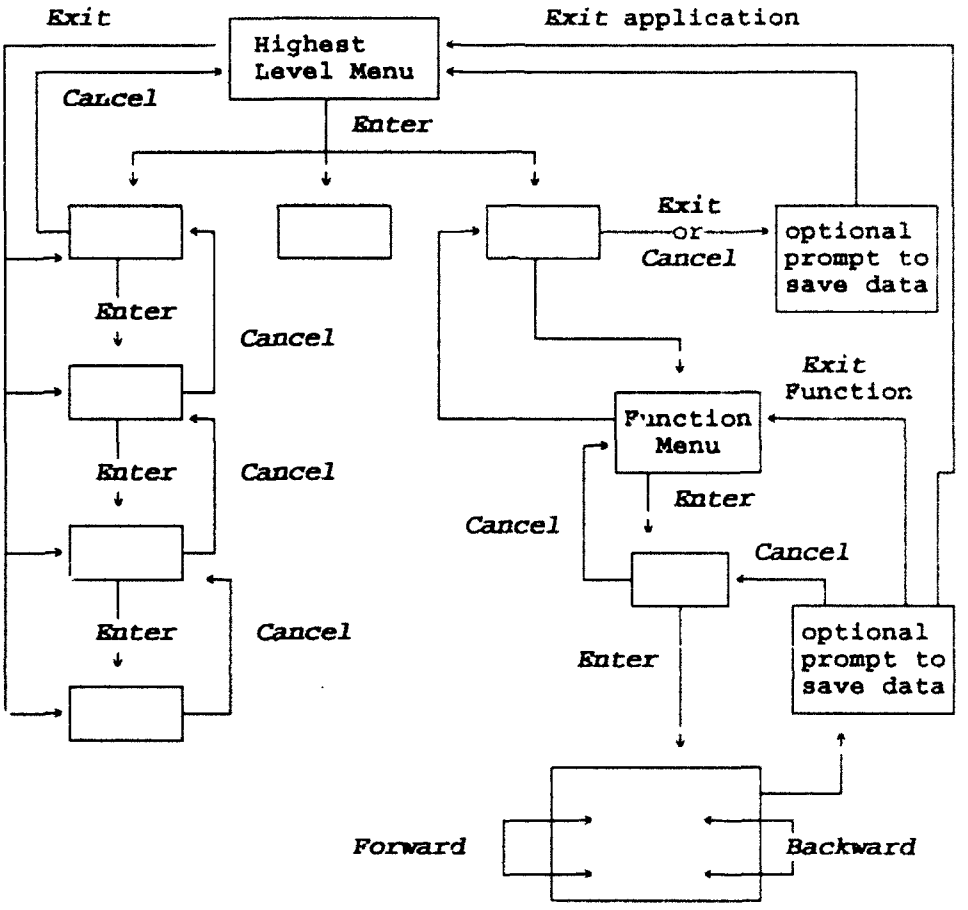
3.1 Dialog Actions

In order to provide a consistent interface, certain dialog actions must have common meanings.

> ***Enter*** - proceed one step (e.g., display a new panel or display current panel with significant changes)
> ***Cancel*** - go back one step (e.g., display previous panel or display current panel with significant changes)
> ***Exit function*** - go back to a specific point in the application
> ***Exit application*** - leave application
>
> ***Backward*** and ***forward*** are not considered navigating actions (e.g., not related to movement among panels) but are scrolling actions (e.g. movement within a panel).

When a user ***cancels*** a panel, information entered on that panel is not saved (although it might be retained as default values for the next time that panel is displayed). If a user's action would cause significant information to be lost, it is required that the dialog prompt the user and ask for confirmation before discarding the information, or allow the user to save the information or to take back the requested action.

3.2 Permissible Navigation Actions



3.3 Windows

An application may run in a ***window environment*** -- that is, panels are displayed in defined and bounded portions of the screen called ***windows***. Computer systems without windowing capabilities have a screen that is regarded as a full-sized window and cannot be moved or changed in size. Systems with windowing capabilities allow users to partition the screen into individual windows, each containing its own panel. By using multiple windows, a user can simultaneously view several panels from the same application or different applications.

If a screen contains two or more windows, a user might not see an entire panel in each window, depending on the size of the window. A user can move and change the size of each window to accommodate whatever information must be seen. A user may also scroll the contents of windows.

Either the operating system (and its presentation tools, services, and window manager) or the application itself provides windowing capabilities.

3.4 Types of Windows

A ***primary window*** is a window in which a user and computer carry out their primary dialog. For example, the primary window in a spreadsheet application is that window that displays the spreadsheet. In systems without windowing capabilities, the full screen is the primary window.

A primary window can contain as many panels as are necessary to carry on the dialog.

A ***secondary window*** comes from a primary window. It is a window in which a user and computer carry on a dialog parallel to the one going on in the primary window. For example, a spreadsheet application may have a secondary window which contains a panel that allows a user to format options for the spreadsheet. Secondary windows may also be used to display help information. A user can switch back and forth between primary and secondary windows.

Primary and secondary windows have title bars at the top of the window. The window title associates the window with its application.

A ***pop-up window*** is a portion of the screen in which a panel is displayed that extends a user's dialog with either a primary or secondary window. One use of pop-ups is for the display of messages.

A user must complete the interaction with a pop-up window before continuing the dialog with the underlying window. A user can move and resize primary and secondary windows, but pop-up windows are of a fixed size and location.

3.5 Input Devices - Keyboard, Mouse, Other

The Finance User Interface Style Guide supports the concurrent use of a keyboard and a mouse (or other pointing device that acts like a mouse). On programmable terminals (PCs, etc.) a user should be able to switch between a keyboard and a mouse at almost any point in a dialog without having to change application modes.

3.6 Keyboards Supported

This document is designed around one keyboard specification, the IBM Enhanced keyboard. The approach for dealing with this keyboard and other types of keyboards will be covered in a later chapter.

3.7 Character and Graphics Application

A ***character application*** is one that limits itself to character-mode elements. This is the focus of this document.

A ***graphics application*** is designed to take advantage of the graphics-mode elements, such as radio buttons, push buttons, etc. These types of applications should use the DoD Human Computer Interface Style Guide instead of this style guide.

An application that mixes character and graphic modes will be considered a graphics application.

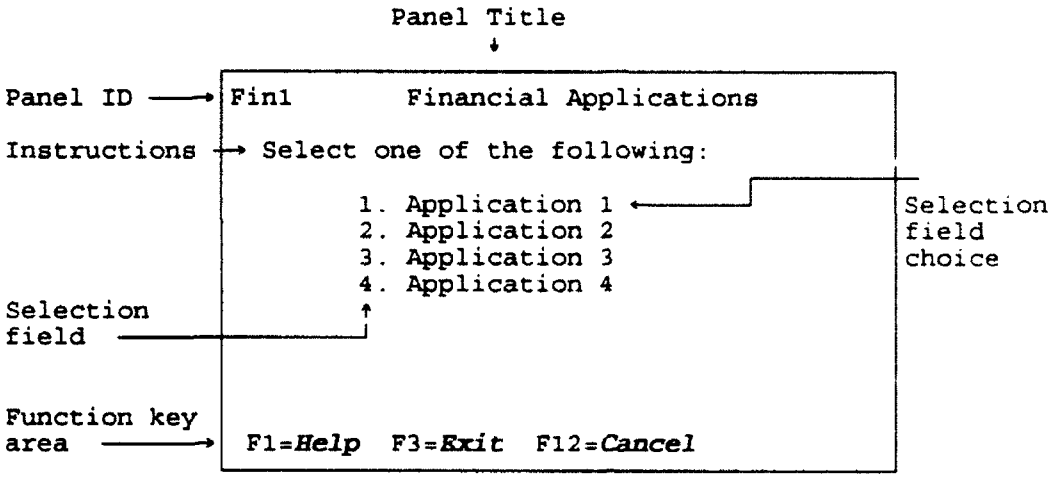
3.8 User Preference

This guide allows applications to give a user some options to customize parts of the interface. This freedom to customize can significantly improve user productivity.

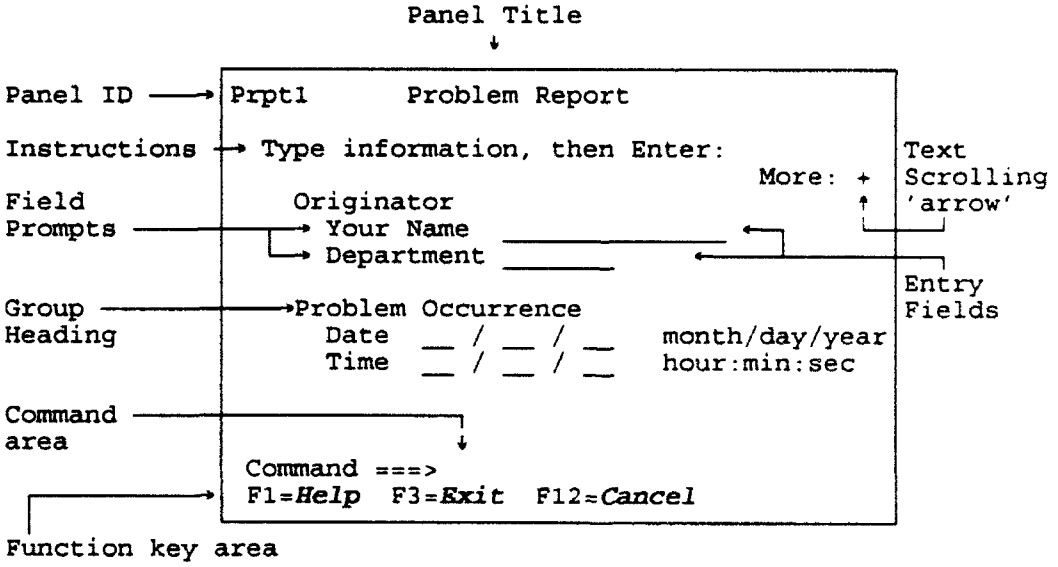
The options a user selects to customize the interface should remain in effect from one application to another (although some options may apply to only one type of application, such as cell size for a spreadsheet application).

3.9 Sample Panel Designs

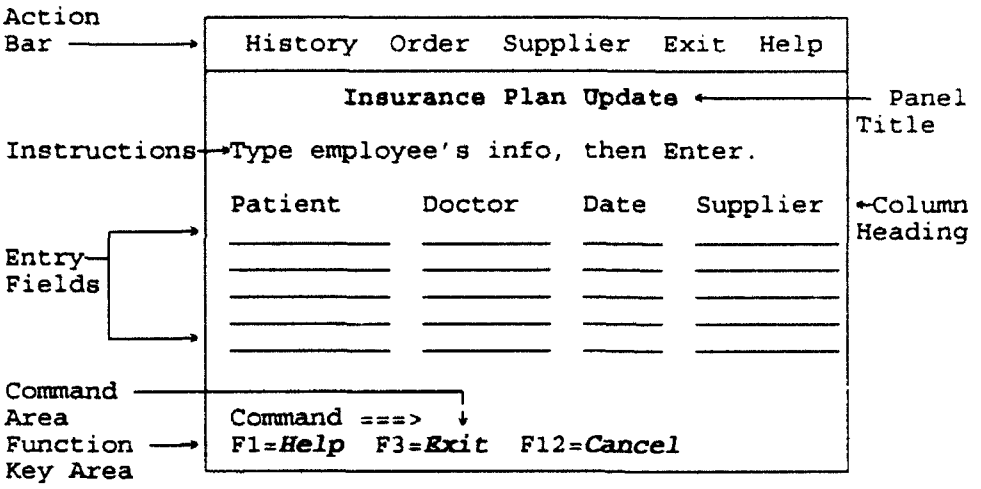
Menu Panel



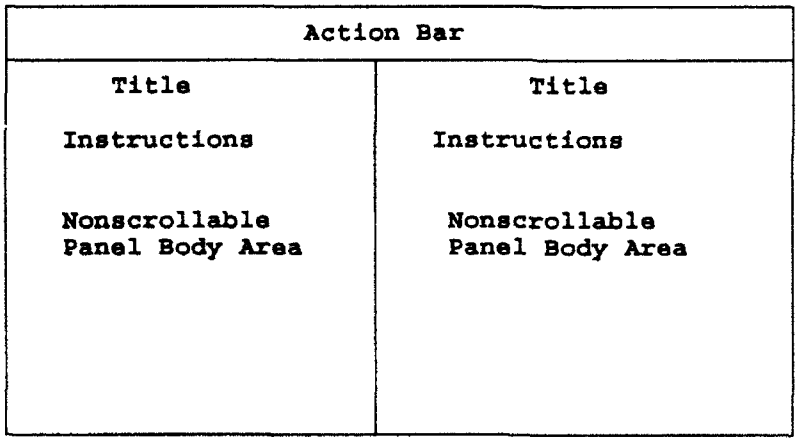
Form Fill-in Entry Panel



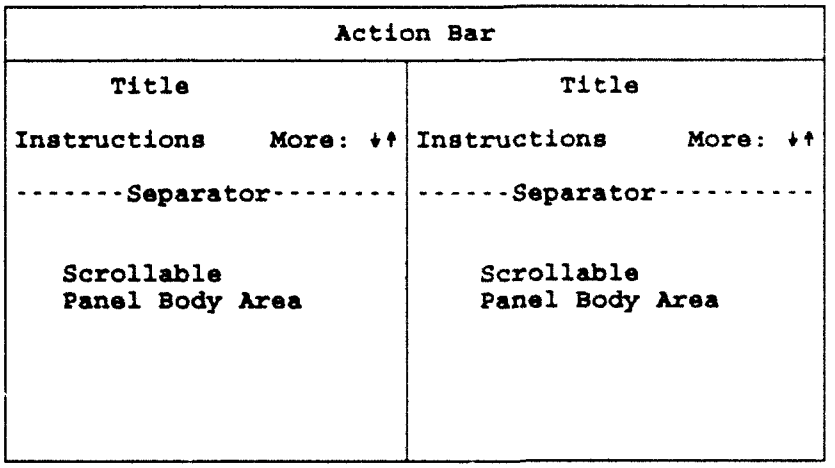
Tabular Entry Panel



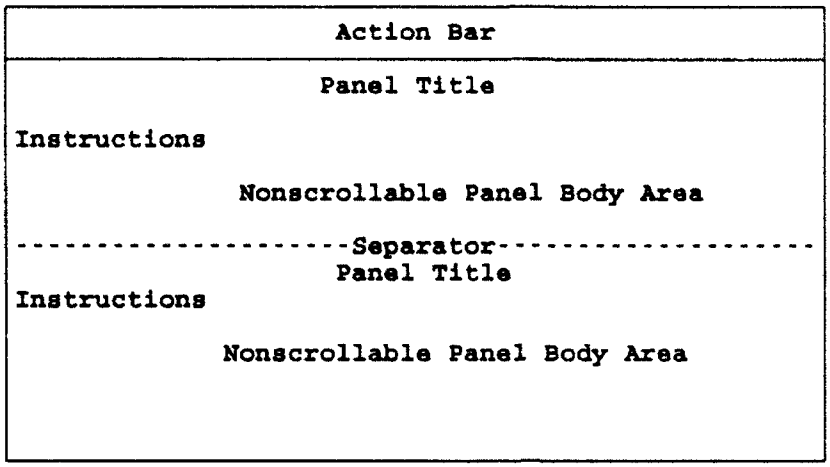
Sample Panel Layouts



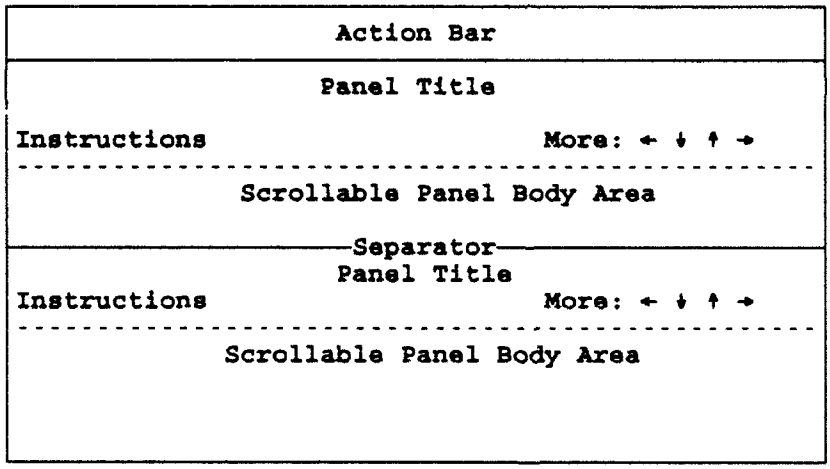
Layout #1: A vertical separator line divides the panel into two vertical panel body areas that do not scroll. Each panel body has its own title and instruction.



Layout #2: A vertical separator line divides the panel itno two vertical panel body areas that scroll independently. Each panel body has its own title instruction, and scrolling arrows (More: ↓↑).



Layout #3: This layout divides the panel into two horizontal panel body areas that do not scroll. Each panel body area has its own title and instructions.



Layout #4: This layout divides the panel into two hoizontal panel body areas that scroll independently. Each panel body area has its own title, instructions, and scrolling arrows (More: ← ↓ ↑ →).

Chapter 4

Selection Fields and Entry Fields

4.1 Selection Cursor

A selection cursor is a form of emphasis used to highlight selection field choices, and entry fields, one at a time. The selection cursor shows the user where on the screen interaction will occur; it is the focal point of a panel. A selection cursor varies from one type of display to another. Sometimes it is an outline box, on another device it may be reverse color. On nonprogrammable terminals the selection cursor is the ***character cursor*** generated by the terminal hardware (it can move outside of entry fields and selection fields). It is the intent of the Finance User Interface Style Guide to use the selection cursor in a ***point-and-select*** technique. That is, the user points to an object on a panel (selection-field choice or entry field) and then selects an action to work on that object. Users move the selection cursor from item to item in a panel by pressing keys or moving a mouse.

4.2 Selection Fields

A selection field is a general term for a set of related choices from which a user makes a selection. The choices in the selection field can be one or more words. To ***select*** a choice from a selection field, a user moves the selection cursor to that choice. The word ***select*** is not synonymous with ***enter***, which means the panel containing the selected choice is submitted to the computer for processing. When a user selects a choice, the application should immediately acknowledge, visually, that the choice was selected. This feedback can be provided by another form of emphasis that uses color, highlighting, or a character/symbol placed in front of the selected choice. The color or highlighting is called ***selected emphasis*** and the character/symbol is called a ***selection indicator***. On nonprogrammable terminals, selected emphasis will not display until a host interrupt is generated to update the screen.

Types of Selection Fields

Single-choice. A user must select only one choice and is required to make a selection.

Multiple-choice. A user can select any number of choices or none at all. A user needs to make an explicit selection.

Extended-choice. A user can: 1. select only one choice when the extended-choice field is in its initial state (e.g., the extended-choice field is a single-choice field that is ***auto-select***); 2. select two or more choices when a user requests extended-choice selection (e.g., the extended-choice field is a multiple-choice field in this state).

An extended-choice selection field is an application option; visual characteristics of the selection field tells a user if extended-choice is available. Extended-choice fields are not available on nonprogrammable terminals, multiple-choice selection fields may be used instead.

Any of the selection field types may be displayed as a ***scrollable selection field***. This approach will have the choices appear in a box in a panel, the user can then scroll the list of choices in the box (without scrolling the entire panel).

4.3 Auto-Select

This special technique simplifies user interaction and may reduce selection errors. Auto-select is used when a panel contains one field—it must be a selection field—from which a user is allowed to select a single choice. When a panel containing an auto-select field is initially displayed, one choice in the field is always selected (default) and displayed with emphasis. The selection cursor is on the selected choice and as a user moves the cursor from choice to choice, the selected emphasis moves with the cursor. When the selection cursor is on the desired choice, the user enters the selection. On nonprogrammable terminals, choices are not displayed with selected emphasis as the cursor is moved.

This technique ensures that the user can select only one choice. Since this technique requires the user to make a choice, a ***none of the above*** choice may be appropriate.

4.4 Auto-Enter

Auto-enter provides an automatic ***entry*** of panels without a separate ***enter*** request by the user. Auto-enter is allowed when the user selects a choice by typing a choice number or mnemonic (or selection was made with a mouse). These types of selection causes a panel to be automatically ***entered*** for processing, thus eliminating the ***enter*** step. Auto-enter can only be used with an auto-select field (a single-choice selection field that is the sole field in a panel or pull-down). Auto-enter is available on some nonprogrammable terminals.

4.5 Numbers and Mnemonics

Selection fields contain entry fields in which a user types numbers or mnemonics to select choices. In single-choice fields, the entry field precedes the upper-left choice. The entry field has many character positions as will accommodate the largest number assigned to a selection field choice. For example, if the selection field contains nine or fewer choices, the entry field is one character long. If the selection field contains 10 to 99 choices, the entry field is two characters long.

In multiple-choice selection fields, each choice is preceded by a one-character choice entry field. A user may type a slash (any other non-blank character may be designated) in the entry field beside each choice to be selected.

4.6 Content Style

Only the first letter of a choice is capitalized, except when it is an acronym or abbreviation that would normally be capitalized.

An ***ellipsis*** (three dots) is placed after any choice that results in a pop-up. There are no spaces between the dots or between ellipsis and the choice text.

4.7 Numbering Choices

Number choices in a selection field will be used when the following are true:

- The field is a single-choice selection field. Do not number multiple-choice selection field choices.
- The field contains nine or fewer choices (if the application uses nonprogrammable terminals, then there is an unlimited number of numbered choices).

If the choices themselves contain numbers, place text between the numbering and the choices. For example:

GOOD	POOR
1. Building 21	Buildings:
2. Building 25	1. 21
3. Building 39	2. 25
	3. 39

Number choices in selection fields sequentially and consecutively beginning with 1 in each field. Use the format: digit, period, space, choice text.

4.8 Mnemonics

A mnemonic is a single highlighted character that provides a fast way of selecting a selection-field choice. A user types the mnemonic to select the choice. Mnemonics will not be used for multiple-choice selection fields.

Displaying mnemonics is a user option (user decides if they are of or on). When a user turns mnemonics off, highlighting disappears and the mnemonic characters do not function. For nonprogrammable terminals the default will be mnemonics off, for programmable terminals the default will be on.

Mnemonics will not be used for fields that vary from one presentation of the panel to the next (such as a listing of file names).

Mnemonics will not be used if the choices are numbered.

4.9 Choosing Mnemonic Characters

Mnemonic characters should be easy for the user to remember.

- The first character of the choice is used, or a character of one of the words in a multiple-word choice, if that character has not been used as a mnemonic for another choice in the current field.

  Example:

  Print
  Page Width
  Page Length

- A consonant in the choice text is used if the character has not been used as a mnemonic for another choice in the same field.

  Example:

  Copy
  Move
  Paste
  Print

  In this example, the mnemonic for ***paste*** is ***p*** because it is the first character of the choice text. The mnemonic for ***print*** is ***r*** because ***p*** has already been used.

- Any other character in the choice text can be used if that character has not been used as a mnemonic for another choice in the current field.

- If all characters in a choice text have been previously used as mnemonics in the current field, a suffix is used.

  Example:

  Open (M)

- Non-alphabetic characters (comma, asterisk, plus sign, minus sign, etc.) may be used if they are meaningful.

- A mnemonic may be used only once in a selection field but may be used in multiple selection fields in the same panel.

Mnemonics are only active in the field containing the selection cursor.

4.10 Scrollable Selection Fields

A scrollable selection field will be used only for a list of choices in a single column (the list may be fixed or variable length). A scrollable selection field contains a title, scrolling information, and a list of choices, all surrounded by a box. A scrollable selection field:

- Includes a title, if a field prompt or heading is not used. Title will be centered in box and separated from the list of choices.

- Has scrolling arrows (up and down) will be right-justified on the same line as the title.

- Will have "unavailable" emphasis for those choices not available from a field with a fixed list of choices.

- Will not display unavailable choices for a field with variable length list of choices.

All choices will be left-aligned within the box. A choice list will contain at least three choices at full size.

4.11 Selection Cursor

The selection cursor is used to select entry fields to type in as well as selection field choices. The selection cursor appears on the screen as a form of emphasis or highlighting, such as reverse color. The selection field choice or entry field the selection cursor is on is highlighted. The selection cursor is present only when a panel contains a selection field or entry field. One selection cursor appears at a time. For nonprogrammable terminals the character cursor is used as the selection cursor for pointing to choices and typing; its appearance depends on the type of terminal.

When the selection cursor is on a selection field, it appears in one of three forms (the following do not apply to nonprogrammable terminals):

- For single-choice (auto-select) fields a reverse color bar is used.
- For single-choice (not auto-select) and for multiple-choice fields, an underline is used
- For an extended-choice field in its initial state a color bar with underline is used (on monochrome screens a reverse color bar is used). When the field changes to multiple choice, the selection cursor changes to an underline.

For choices aligned in a single column, the selection cursor is the length of the longest choice in the selection field, plus one leading and one trailing blank.

For choices aligned horizontally on the same line, or in several rows, the selection cursor will be the same length as the choice the cursor is on plus one leading and one trailing blank.

4.11.1 Selection Cursor Initial Position

When a selection field is the first field in a panel or when a user tabs into a selection field, the initial position of the selection cursor in the field is on the topmost and left-most choice. If an auto-select field contains a default choice other than the first choice, the initial position of the selection cursor will be on the first default choice (not always possible with nonprogrammable terminals).

4.12 Selection Field Selected Emphasis

When a user selects a choice from a list, its appearance must be emphasized to remind the user that it has been selected. Selected emphasis is used, where physically possible, by changing the color or highlighting of the selected choice. (See 13.0, "Color and Emphasis.")

When a user selects a choice and enters a panel, the choice appears with selected emphasis if the panel remains displayed.

Default choices are displayed with selected emphasis when the panel initially appears.

4.13 Selection Field Indicators

Indicators can also be used to emphasize selection. Indicators are especially useful for nonprogrammable terminals with limited color/highlighting capabilities.

Selection field indicators will be used in the following ways:

- For single-choice, auto-select fields, the selection indicator is the selection cursor, a reverse color bar. A solid, right-pointing triangle (or greater-than symbol) is used to indicate the selected choice. The symbol does not move when the user moves the selection cursor to a different choice. The next time the field is displayed, the symbol appears with the new choice. The symbol appears in the second position to the left of the choice text. The selection cursor does not cover this symbol. For nonprogrammable terminals, the selection indicator may be the text cursor or it may be the number or mnemonic in the choice entry field.

- For multiple-choice selection fields, and single-choice selection fields that are not auto-select, use a right-pointing triangle (or greater-than symbol) is used. The symbol appears in the second position to the left of the choice text. The selection cursor does not cover this symbol. For nonprogrammable terminals the selection indicator for single-choice is the number or mnemonic a user types in the choice entry field. On nonprogrammable terminals for multiple-choice, the slash (or possibly another character) that a user types in is the selection indicator.

4.14 Selection Field Unavailable Choice Emphasis

A choice is unavailable if the current state of an application does not allow a user to select it. A user must be made aware through emphasis if choices in various panel elements are unavailable.

Unavailable choices can be de-emphasized by reducing the contrast between the choice and the background so that unavailable choices stand out less than available choices. If this technique is not possible, some color displays change the color of unavailable choices, so that they are de-emphasized compared to the color of available choices.

For monochrome displays, the first non-blank character of the unavailable choice can be replaced with an asterisk. If the choices are numbered, the asterisk can be placed over the first digit of the number of the choice.

The unavailable emphasis shows a user which choices may not be selected. No emphasis is needed for unavailable choices. A user will be able to request help for an unavailable choice. A message will display if a user attempts to select an unavailable choice.

4.15 User Interaction With Selection Fields

Selection Field Initial Conditions

When a panel is initially displayed, default choices are already selected and displayed with selected emphasis. Single-choice auto-select fields always appear with a default choice because it is the choice on which the cursor is initially positioned.

The default choice of a selection field is one of the following:

- The choice that was selected during a previous appearance of the field.
- A program-selected choice the first time a selection field appears during the current execution of the program or, to reflect changes in the state of the program.

Selection Field Initial State

A selection field may contain choices that represent the current state of some part of the application. When a selection field is displayed to show current state, an indicator symbol will be used to indicate the state currently selected. For nonprogrammable terminals, the selected current state is indicated by the number or mnemonic in the entry field.

4.16 User Interaction Using a Keyboard

The following methods show how a user uses a keyboard to move the selection cursor within selection fields.

Once the selection cursor is in a selection field, a user presses the arrow keys to move the cursor from choice to choice. For nonprogrammable terminals:

- In a single-choice selection field a user may tab only to the choice entry field.
- In a single-choice, auto-select field, a user may press the arrow key to move the cursor to a choice.
- In a multiple-choice selection field, a user tabs to the entry field to the left of each valid choice. A user may also use the arrow, tab, backtab, and new line keys to move the cursor between choices.

Moving the Selection Cursor by Typing Letters

In fields containing a variable list of choices (e.g. a list that may change from one presentation of the panel to the next), a user can type a letter to move the selection cursor to the next choice that starts with that letter. The selection cursor always moves forward just as though the Down Arrow key had been pressed. If no choice matches the letter typed, the cursor does not move and a "beep" sounds, with a message displayed by the application. The message is removed with the next user action. This approach is not available on all nonprogrammable terminals.

Scrolling a Scrollable Selection Field Using a Keyboard

A user may scroll a scrollable selection field by moving the selection cursor with the arrow keys. The list of choices in a scrollable selection field scrolls when the cursor reaches the top or bottom of the box. When the cursor reaches the beginning or end of the list of choices, the cursor does not wrap; scrolling and cursor movement stop.

Use the forward and backward actions for cursor-independent scrolling while the cursor is in the scrollable selection field. These actions scroll the field an amount equal to the number of choices displayed minus one. Backward makes choices toward the top of the list visible; forward makes choices toward the bottom of the list visible. For nonprogrammable terminals the user can scroll selection fields with the forward and backward actions, but the arrow keys do not make the selection field scroll.

Along with arrow keys, the following functions are also supported in scrollable selection fields:

- ***Beginning of data***—moves the selection cursor to the first choice in the scrollable selection field, scrolling the list of choices if necessary.
- ***End of data***—moves the selection cursor to the last choice in the scrollable selection field, scrolling the list choices if necessary.
- ***Tab and backtab***—move the selection cursor to the next and previous fields in the panel.
- ***Beginning of line***—moves the cursor to the first field on the same line.
- ***End of line***—this moves the selection cursor to the last field on the same line as the cursor in the scrollable selection field.

Selecting a Choice Using a Keyboard

A user can select a choice in a selection field several ways

- In a single-choice field a user can:
  - Move the selection cursor from choice to choice in an auto-select field. Every time the cursor moves to a new choice, it is automatically selected.
  - Type the number of the choice if the choices are numbered. If the single-choice field is the only one in the panel, or pull-down, the panel is automatically entered (not on some nonprogrammable terminals). If the panel, or pull-down, contains more than one selection field, the field with the selection cursor in it is active. When a user types the number, the choice is selected. If a user types a number that is not in the selection field, a "beep" sounds (except on some nonprogrammable terminals). If the panel contains a message area, it is an application option to display a message explaining the error. The message is removed by the next user action.
  - Type the mnemonic of the choice if the choices have mnemonics. Mnemonics can be typed in either upper- or lowercase. If the single-choice field is the only field in the panel, the panel will automatically be entered (not on some nonprogrammable terminals). If a user types a character that is not a mnemonic in the field, a "beep" sounds (not on some nonprogrammable terminals). If the panel contains a fixed message area, it is an application option to display a message explaining the error. The message is removed by the next user action.
  - Move the cursor to the choice and press the spacebar or the slash character if the field is not auto-select (not on nonprogrammable terminals).
- In a multiple-choice field, a user can move the selection cursor to the desired choice and press the spacebar (not on nonprogrammable terminals) or slash character.
- In an extended-choice selection field, the choice that currently has the cursor on it is selected. If the user wants to select a different choice, the cursor must be moved to that choice and it will show selected emphasis for auto-select. To select multiple choices, the user must type a slash character or press the spacebar while the selection cursor is on one of the desired choices.

The selected emphasis appropriate for an auto-select choice disappears and a selection indicator appears. The appearance of the selection cursor changes to indicate to the user that multiple choices can be made. The user moves the selection cursor to the next choice and presses the spacebar or slash character to select that choice.

Cancelling a Selection Using a Keyboard

A user can cancel a choice in an auto-select field by moving the selection cursor to a different choice, including the *none* choice if all other choices are optional.

In a single-choice field that is not auto-select, a user can cancel a selection by selecting a different choice. This is done by typing the number or mnemonic of the new choice, or by placing the cursor on the new choice and either pressing the spacebar or typing a slash character.

A user cancels a multiple-choice field by pressing the spacebar while the cursor is on the selected choice.

In an extended-choice selection field that is in its single-choice, auto-select mode, a user can cancel a choice by moving the cursor to a different choice. When the selection field is in its multiple-choice mode, a user may press the spacebar while the cursor is on a selected choice to cancel the selection. When there is only one selected choice remaining, the cursor changes appearance to indicate that the field is in its initial state as a single-choice auto-select field.

For nonprogrammable terminals, a user may make a change before the screen is "entered" to the computer for processing.

4.17 User Interaction Using a Mouse

Moving the Selection Cursor

One way that a user can move the selection cursor is to hold down the ***select button*** while moving the mouse. It will give the appearance that the mouse pointer is "dragging" the selection cursor. The user may also move the mouse with the select button released. The selection cursor will remain stationary until the mouse pointer is on the desired choice and the user presses the select button. The selection cursor then jumps to that choice.

Scrolling a Scrollable Selection Field Using a Mouse

A mouse user can scroll a scrollable selection field by placing the mouse pointer in the selection cursor, holding down the select button and dragging the selection cursor from choice to choice. The list choices scrolls when the user continues to move the mouse after the cursor reaches the top or bottom of the box. The cursor does not wrap.

Selecting a Choice Using a Mouse

A user can select a choice by pressing the select button, moving the mouse pointer to the desired choice, and releasing the select button. A user may also move the pointer to the desired choice and press and release the select button.

An extended-choice selection field in its single-choice mode is changed to its multiple-choice mode when a user presses and hold the Ctrl key while pointing at and selecting a choice with a mouse. Once a second choice has been selected, a user no longer has to hold down the Ctrl key to make additional selections. A user can select several choices that are next to each other by holding a Shift key and the mouse select button and dragging the mouse pointer across the desired choices. If the choices have already been selected, this technique can also be used to cancel selections. If a user does not hold down a Shift key, the selection cursor moves over the choices but choices are not selected or, if already selected, their selection is not canceled.

Fast-path Selection for Mouse Users

A fast-path method to select choices and submit the panel is: the user presses and releases the select button twice (double click) while the mouse pointer is on a choice in the panel body, to both select the choice and perform an application-specific action on the choice.

This approach should only be used when:

- The action bar contains multiple actions that affect the choices in the panel body, otherwise a single click causes the action.
- The panel body contains only one selection field.
- The selection field is a single-choice field or an extended-choice field in single-choice mode.

The panel title or instruction will identify the default action.

Canceling a Selection Using a Mouse

A user can cancel a selection in a single-choice auto-select field by moving the cursor to a different choice.

In a single-choice field that is not auto-select, a user may cancel a selection by selecting a different choice. In a multiple-choice field, a user cancels a selection by pressing the select button when the mouse pointer is on the selected choice to be canceled and then releasing the select button.

In an extended-choice selection field that is in its multiple-choice mode, a user cancels a selection by repeating the selection action on an already selected choice. When there is only one selected choice remaining, the selection cursor changes appearance to indicate that the field is in its initial state as a single-choice auto-select field.

4.18 What Happens When an Available Choice Is Selected

When a user selects an available choice from a selection field, the choice is displayed with selected emphasis, even if the panel will be immediately replaced, overlaid, or removed. The selected emphasis tells the user that the selection was made. On nonprogrammable terminals, the selected emphasis will not appear until a host interrupt is generated to update the screen. If the panel on which a user selected a choice is redisplayed after an interrupt, the choice appears with selected emphasis until the selected action is complete. Whatever else happens following the selection depends on the application.

4.19 What Happens When an Unavailable Choice Is Selected

When a user selects an unavailable choice, a pop-up message appears telling the user that the choice is not available. The message also reminds the user to request help for that choice. The help panel tells the user the conditions that make the choice unavailable.

4.20 Selection Field Examples

```
                    Title

Select one of the following:

  1. Bold
  2. Underline
  3. Uppercase
  4. Section Numbers
  5. None



F1=Help  F12=Cancel
```

Example #1

Single-choice, auto-select selection field interaction. The "current-state" choice previously selected by the user was ***underline*** as indicated by the reverse bar. The user can either accept this default by requesting an ***enter***, or select another choice. The up and down arrow keys are used to select another choice.

Alternatively, a user can make a selection by typing one of the mnemonics (underlined characters) or the number of the desired choice. In both cases, the selection cursor moves to the choice. In both cases, the selection cursor moves to the selected choice and entered for processing (since this is an auto-select panel).

A mouse user would place the mouse pointer over the desired choice and press the select button. The selection cursor would move to the selected choice, the choice is selected, and when the button is released the panel is entered (auto-select). If the select button is held down and the pointer moved over any other choices, the selection cursor moves (not all nonprogrammable terminals) to the choices pointed to by the pointer.

```
                    Title

Select one or more of the following:

     Bold
  X  Underline
     Uppercase
  X  Section Numbers



F1=Help  F12=Cancel
```

Example #2

Multiple-choice selection field interaction. All of the choices of this multiple-choice selection field are available. The "current-state" selected choices when this panel is first displayed are ***underline*** and ***section numbers*** (they both have an *X* indicator). The selection cursor initially appears on ***bold***.

To cancel both selected choices, the user would press the Down arrow key, then the spacebar to cancel ***underline***. Then press the Down arrow twice to move to ***section numbers*** and again press the spacebar.

The user requests an ***enter*** to submit panel with desired selections.

```
                    Title

Select one of the following:

_ 1. Bold
  2. Underline
  3. Uppercase
  4. Section Numbers



F1=Help  F12=Cancel
```

Example #3

Nonprogrammable terminal single-choice selection field. When this panel is initially displayed, the selection cursor is on the entry field next to the ***bold*** choice.

A user might select ***section numbers*** by typing a ***4*** and then requesting an ***enter*** to submit the panel. Or, the user might move the cursor to ***section numbers*** by pressing the Down arrow key three times and then requesting enter.

```
                    Title

Select one or more of the following.
Then Enter.
_ 1. Bold
/ 2. Underline
_ 3. Uppercase
/ 4. Section Numbers



F1=Help  F12=Cancel
```

Example #4

Nonprogrammable terminal multiple-choice selection field. When this panel is initially displayed, the default choices are ***underline*** and ***section numbers*** (each has a slash next to it).
The cursor initially appears on the first choice in the panel, ***bold***.

A keyboard user can cancel the two selected choices by first pressing the ***new line*** key once to move the cursor to the entry field in front of the ***underline*** choice. When the user then presses the spacebar, the choice selection is canceled. The user then presses the ***new line*** key twice more to move to the ***section numbers*** choice and presses the spacebar to cancel that choice. Any choice is selected by typing a slash (/) next to the desired choice. The user submits the panel by requesting enter.

4.21 Entry Fields

An entry field is a space in a panel in which a user may type information. The location and layout of left-to-right entry fields is covered in this section.

4.21.1 Entry Field Appearance

Scrollable entry fields are an application option, but can only be used if the total length of information exceeds 25 characters; the visible portion of the information must be at least 10 characters.

When a panel first appears, an entry field is displayed either blank or with an initial value. Entry fields may contain separator characters, such as hyphens to separate digits into three groups as in a social security number. For nonprogrammable terminals, this type of multi-part entry field may be created in two ways:

1. Use a protected field, consisting of two attribute bytes and a separator character, to separate individual parts of a multi-part entry field.

2. Use only attribute bytes, not separator characters, to separate individual entry fields in a multi-part entry field. The attribute bytes appear as blanks. The attribute byte for each part should cause automatic tabbing at the end of that part to the next part.

Nonscrollable Entry Fields

Use the underscore field attribute (if supported by the terminal) to designate all available entry positions in an entry field. For displays that do not support the underscore attribute, place left and right brackets at either end of the available entry positions in an entry field. The brackets indicate that no additional entry positions exist beyond the bracket and, therefore, the entry field does not scroll. For terminals that do not support the underscore attribute or brackets, use the underscore character. Characters typed in by the user will then replace the underscore characters.

Scrollable Entry Fields

Entry fields that typically have short entries but, occasionally, require longer ones, are candidates for scrollable entry fields. Use the greater-than and less-than symbols at the ends of the entry field to indicate that more field space is available in the direction indicated by the symbol. The greater-than symbol (>) indicates more space is available to the right; the less-than symbol (<) indicates more space is available to the left. The direction symbols will always be at the extreme ends of the field. If the display does not support the underscore attribute, and brackets are used, the less-than and greater-than symbols replace the appropriate bracket.

4.21.2 Entry Field Examples

Example #1

```
Employee's name : Smith, James_   >
                  ---------------
```

Employee's name : is a field prompt. The underscored area represents the entry field. The current entry field value is ***Smith, James***. The > indicates that the entry field scrolls right.

Example #2

Employee's name : <th, James_>

The < and > indicates that this entry field scrolls left and right.

Example #3

Employee's name : <, James _

Same as example #2 except that the field can not be scrolled any further to the right (the > symbol has disappeared). User may scroll to the left (<).

Example #4

Employee's name : <th, James]

This entry field scrolls left but not right. The terminal does not support the underscore attribute, so a bracket (*]*) is used to indicate the end of the field.

Example #5

Employee's name : Smith, James_

This entry field is not scrollable.

Example #6

Social Security Number: _ — —

This field contains two separator characters (-). As a user types in the SSN, the text cursor automatically skips over the separator characters.

Social Security Number: 123-_ -

After the user enters the numbers ***123***, the cursor automatically jumps over the separator, waiting for the fourth number to be typed.

4.21.3 Entry Field Text Cursor

A text cursor is a symbol that indicates where typing input will occur. It appears only in entry fields, when the selection cursor is positioned on an entry field. If a user moves the selection cursor to an entry field using a mouse, the text cursor appears.

On nonprogrammable terminals, the text cursor is a single-character cursor, the same as the selection cursor, and remains the same in replace mode or insert mode.

4.22 Text Cursor Appearance

In replace mode the cursor is an underscore; in insert mode it is a reverse color box. For nonprogrammable terminals, the user may select an underscore or a solid block and does not depend on whether the user is typing in replace or insert mode.

4.23 Text Cursor Location

When a user moves the selection cursor to an entry field, the default text cursor position in replace mode is the first character position (the left-most position in a left-aligned entry field). If the user is in an insert mode, and the entry field contains information, the default text cursor position is in the position following the last character in the field.

In both replace and insert modes, it is an application option to position the cursor in a more useful position within the field.

For a mouse user, the text cursor appears where the mouse pointer is pointing in an entry field when the select button is pressed. If the pointer is not pointing to a character in the entry field, or, if the user is holding down the button to move the selection cursor, the text cursor appears in the first position for typing information in the entry field.

For nonprogrammable terminals, when a user tabs into an entry field, the cursor appears in the first position (in both replace and insert modes).

4.24 Marking Text

A mouse user can mark text in the entry field by pressing the select button and moving the mouse pointer across the text that is to be marked. Marked text can be deleted and replaced by new text. Not available on nonprogrammable terminals.

4.25 Entry Modes

A user may type information in two modes; replace mode or insert mode. The Insert key is used to toggle between the two modes. Users of nonprogrammable terminals must use the Reset key to change from insert mode to replace mode.

When an application is initiated, the default mode is replace. The look of the cursor tells a user if insert or replace mode is in effect, except on nonprogrammable terminals (which use a status indicator).

Except for nonprogrammable terminals, a single-character field will have a replace mode only.

Replace Mode Text Cursor Characteristics

An underscore cursor shows the current entry position in replace mode (nonprogrammable terminals use status indicator). The following rules apply to the text cursor in replace mode:

- When a user types a character, the new character replaces the existing character in that position. The text cursor then moves right to the next entry position.
- The cursor stops when it reaches the last entry position.
- If the field scrolls, a user can type more information each time the field scrolls. The information will continue to scroll until the cursor reaches the last entry field position. The cursor stops when the last character of information is in the last entry filed position unless the field is an auto-tab field (see "The Auto-tab Application Option"), in which case the cursor skips to the next entry field. On nonprogrammable terminals, the text cursor moves to the position following the entry field when the last entry field position is filled unless the field is an auto-tab field; the field does not scroll during typing.
- If a user overtypes a character in the last information character position, a "beep" will sound (not for all nonprogrammable terminals). If the panel contains a message area, it is an application option to display a message explaining the error. The message is removed by the next user action. On nonprogrammable terminals, further input is inhibited (input-inhibited indicator appears) until the user presses the Reset key.
- The Backspace key deletes the character to the left of the cursor. On nonprogrammable terminals, the Backspace key moves the cursor left one position but does not delete a character.
- The text cursor skips over separator characters to the next entry position while a user types. In entry fields that scroll, skipping can make the information scroll more than one character at a time.

Insert Mode Text Cursor Characteristics

The following rules apply while a user is in insert mode:

- When a user types information, the cursor and all characters to the right of the cursor shift one position at a time to the right. The new character appears in the vacated position.

- If the user tries to type a character in a full entry field, a "beep" sounds (not all nonprogrammable terminals) and the character is not entered. If the panel contains a message area, it is an application option to display a message explaining the error. The message is removed by the next user action. On nonprogrammable terminals, further input is inhibited (the input-inhibited indicator appears) until the user presses the Reset key.

- The text cursor skips over separator characters to the next entry character position while a user types. In entry fields that scroll, skipping can make the information scroll more than one character at a time.

4.26 What Happens When an Invalid Value Is Typed In

If a user types an invalid value in an entry field, the value will be shown with error emphasis and a message will appear. The message indicates that the value is invalid and that help is available for that entry field. A help panel should describe the valid values for the entry field.

4.27 Entry Field Scrolling

Entry fields can scroll or not scroll, it is an application option. Scrollable fields will be used only for those fields that have a total length of information characters exceeding 25 characters. The scrollable field must display a minimum of 10 characters of information.

4.28 Cursor-Driven Entry Field Scrolling

This section covers cursor-driven scrolling (scrolling that occurs when a field is scrollable and the user types or presses the arrow keys with the cursor at either boundary).

Scrolling Increment While Typing

When a user types, information scrolls one-third the size of the visible part of the entry field (this is the ***scrolling increment***). If the number of positions remaining in the field is less than one-third of the visible field, the field scrolls so that the last input position for the field is in the last visible position.

Scrolling Increment While Moving the Cursor Using Arrow Keys

When a user moves the cursor in an entry field by pressing the arrow keys, the information scrolls one character position for every keystroke at the boundary. The

scrolling increment for arrow keys (not for typing characters) is one position per keystroke. Here is what happens:

- The user moves the cursor with normal cursor movement techniques to position the cursor at the first character of a field that contains information.

- The user presses the Right arrow key to move the cursor right. Every time the user presses the Right arrow key, the cursor moves one character right. If the user presses and holds the Right arrow key, the cursor continues to move right.

- When the cursor reaches the rightmost visible position of the entry field, the following happens:

  - If more information exists to the right of the visible entry field, the visible portion of the field appears to move left one character position.

  - If the field is not completely filled with information, and if the cursor is in the first blank position after the last visible character, the right entry-field delimiter remains visible. The next time the user presses the Right arrow key the cursor moves to the next entry field as though the user had pressed the Tab key.

  - If the field is filled with characters, and the cursor reaches the last visible character, the next time the user presses the Right arrow key, the cursor moves to the next entry field as though the user had pressed the Tab key.

    If, after scrolling right, the user moves the cursor left and scrolls left to the first cahacter position, the next time the user presses the Left arrow key, the cursor jumps to the previous entry field as though the user had pressed the Backtab key.

Scrolling Increment While Moving the Cursor Using a Mouse

A user places the mouse pointer on the greater-than or less-than symbols and presses the select button to scroll one character. If the user holds the button, scrolling continues. The cursor remains stationary while the information scrolls.

4.29 Cursor-Independent Entry Field Scrolling

Cursor-independent scrolling is caused by the left and right actions, not cursor movement. A field can be scrolled this way only when the cursor is in it. If the cursor is not in an entry field that scrolls, the left and right actions may cause panel area scrolling. When a user scrolls in this manner, the text cursor stays in the same place in the field as the field scrolls.

If the number of positions remaining in the field is less than one-third of the visible field, the field scrolls so that the last input position for the field is in the last visible position.

4.30 The Auto-tab Application Option

A field is Auto-tab if, when a user types a character in the last information position of an entry field, the selection cursor tabs to the next field (as if the user had pressed the Tab key).

If the next field is a selection field instead of an entry field, the selection cursor is on the choice specified for the selection field. However, if the next field is another entry field, the text cursor appears in it.

The Auto-tab is an application option and should be used:

- For high-volume, data-entry panels
- For fields that are usually filled with data
- For fixed-length fields.

4.31 Auto-Clear

Auto-clear is a function that clears an initial value in an entry field when a user types the first character of new information. The text cursor must be in the first information character position when the user presses the first key to type a character. The typed character is displayed and all other character positions are cleared. The user then continues typing as usual. This works the same in both replace and insert modes.

Separator characters are not affected by auto-clear.

- Auto-clear is an application option.
- Auto-clear is not available on most nonprogrammable terminals.

4.32 How Various Keys Work in Entry Fields

Left Arrow Key

The Left arrow key moves the text cursor one character position to the left until it reaches the entry field boundary (e.g. until it reaches the first character). If the field scrolls, the Left arrow key moves the cursor to the beginning of the field, while information scrolls right. Pressing the Left arrow key with the cursor at the leftmost entry position has the same effect as pressing the Backtab key. The cursor jumps to the previous field.

When the field contains separator characters, pressing the Left arrow key skips the cursor over the characters. In fields that scroll, this skipping action may make the information scroll more than one character at a time. During this type of scrolling, the next cursor position always comes into view, even if the cursor has to skip over several separator characters.

On nonprogrammable terminals, the Left arrow key moves the text cursor one character position to the left anywhere on the screen. When the cursor reaches the field boundary, pressing the Left arrow key does not scroll the entry field.

Right Arrow Key

The Right arrow key moves the cursor one character position to the right until it reaches the last visible position. If the field scrolls, the Right arrow key moves the cursor until it reaches the next available character space after the end of the information in the field, at the same time information scrolls to the left. Pressing the Right arrow key when the cursor is at the rightmost entry position has the same effect as pressing the Tab key. The cursor jumps to the next field.

When the field contains separator characters, the Right arrow key makes the cursor skip over the characters. In fields that scroll, this skipping action may make the information scroll more than one character at a time. During this type of scrolling, the next cursor osition always comes into view, even if the cursor has to skip over several separator characters.

On nonprogrammable terminals, the Right arrow key moves the text cursor one character position to the right anywhere on the screen. When the cursor reaches the field boundary, pressing the Right arrow key does not scroll the entry field.

Word Left Key

The Word Left key moves the cursor to the first character of the word preceding the nearest blank to the left of the line. Not available on all nonprogrammable terminals.

Word Right Key

The Word Right key moves the cursor to the first non-blank following the nearest blank to the right of the line. Not available on all nonprogrammable terminals.

Delete Key

The Delete key removes the character at the cursor position. Characters to the right of the deleted character move left to fill the vacated position. The cursor remains stationary at the point where the character was deleted.

Backspace Key

The Backspace key deletes the character to the left of the cursor and moves the cursor one space left. The characters to the right of the cursor move left one position to fill the vacated space. The Backspace key has no effect when the cursor is at the beginning of an entry field.

When the entry field contains separator characters, the cursor skips over the characters as the Backspace key is pressed. When the cursor is on a position immediately right of a separator character, pressing the Backspace key deletes the character immediately left of the separator character.

On some nonprogrammable terminals, the Backspace key may work the same as the Left arrow key (moves the text cursor one space left and does not delete the character to the left of the cursor).

Insert Key

The Insert key toggles the mode between replace and insert modes. For nonprogrammable terminals the Insert key is not a toggle key, it only turns insert mode on. The Reset key turns insert mode off.

Erase-to-End-of-Field Key

This key deletes the character at the cursor position and all characters to the end of the field.

Beginning-of-Field Key

The Beginning-of-Field key moves the cursor to the first entry position in the field. The information scrolls if the first character position is not already visible. This key is not available on all nonprogrammable terminals.

End-of-Field Key

This key moves the cursor to the next usuable position following the last caharcter in the field. The information scrolls if the next blank entry position is not visible. If the field is full, the cursor moves to the last character position. This key is not available on all nonprogrammable terminals.

Chapter 5

Moving the Selection Cursor

Much of a user's interaction with an application involves moving the selection cursor. This section covers the methods for moving the selection cursor from choice to choice and field-to-field within a panel.

The selection cursor can move only between selection fields and entry fields except on nonprogrammable terminals, where the cursor moves to any position on the screen when the user presses the arrow keys.

5.1 Direct Cursor Movement—the Arrow Keys

The four arrow keys on a keyboard move a selection cursor directly to selection choices, or entry fields, without a user having to understand the path determined by the layout of fields in the panel.

5.2 Contextual Cursor Movement—the Tab and Backtab Keys

The Tab and Backtab keys move the selection cursor to the next field in a panel by skipping the cursor over any choices in the current field. This gives the user a fast way to move to the next or previous field after a selection in the current field has been made.

The Tab and Backtab keys on nonprogrammable terminals move the cursor to the next or previous unprotected input position, not to the next or previous selection field. That Tab key moves the cursor left-to-right and top-to-bottom while the Backtab key moves the cursor right-to-left and bottom-to-top.

5.3 Arrow Keys

Left Arrow Key

When the selection cursor is on a selection field choice, or when the text cursor is at the beginning of an entry field, the Left arrow key moves the selection to the next item to the left of its current position. When the selection cursor reaches the left most item, it wraps to the rightmost item on the line above. When the cursor reaches the leftmost item on the top line, it wraps by jumping to the rightmost item in the bottom line.

Right Arrow Key

When the selection cursor is on a selection field choice, or when the text cursor is at the right boundary of an entry field, the Right arrow key moves the selection to the next item to the right of its current position. When the selection cursor reaches the rightmost item, it wraps to the leftmost item on the line below. When the cursor reaches the rightmost item on the bottom line, it wraps to the leftmost item in the top line.
Up Arrow Key.

The Up arrow key moves the selection cursor to the next item in the line above. At the top line, the cursor wraps to the bottom of the next column to the left of its current position. At the top of the first column in a panel, the cursor wraps to the last item at the bottom of the last column.

Down Arrow Key

The Down arrow key moves the selection cursor to the next item in the line below. At the bottom line, the cursor wraps to the top of the next column to the right of its current position. At the bottom of the last column in a panel, the cursor wraps to the first item at the top of the first column.

On nonprogrammable terminals, the cursor moves freely to any position on the screen when the user presses the arrow keys.

5.4 Tab and Backtab Keys

Tab Key

The Tab key moves the selection cursor to the next field (either the next entry field or the first choice of the next selection field). The Tab moves the cursor left-to-right and top-to-bottom. At the bottom and rightmost field, the Tab moves the selection cursor to the top and leftmost field in the panel body area. For nonprogrammable terminals, the application can exercise some control over the Tab key movements by making use of field protection attributes.

Backtab Key

The Backtab key moves the selection cursor to the previous field (either the previous entry field or the last choice of the previous selection field). The Backtab moves the cursor right-to-left and bottom-to-top. At the top and leftmost field, the Backtab moves the selection cursor to the bottom and rightmost field in the panel body area. For nonprogrammable terminals, the application can exercise some control over the Backtab key movements by making use of field protection attributes.

5.5 Other Cursor-Movement Functions

The following are not available on some nonprogrammable terminals.

Beginning of Line

This moves the selection cursor to the leftmost position on the current line. For menu and parameter entry panels, the beginning of the line is the leftmost entry field or selection field choice on the line. The application defines leftmost for other panels.

End of Line

This moves the selection cursor to the rightmost position on the current line. For menu and parameter entry panels, the end of the line is the rightmost entry field or selection field choice on the line. The application defines rightmost position for other panels.

Beginning of Data

The beginning-of-data function moves the selection cursor to the leftmost position on the topmost line of data. For menu panels and parameter entry panels, the beginning of data is the leftmost entry field or selection field choice on the topmost line of the panel that contains an item on which the cursor may be position. The application defines the beginning of data for other panels.

Similar keys on nonprogrammable terminals are:

- First-field-on-screen. Moves the cursor to the first entry field on the screen (***home*** key).
- New Line. Moves the cursor down to the first entry field on the next line that contains an entry field.

End of Data

The end-of-data function moves the selection cursor to the rightmost position on the bottommost line of data. For menu panels and parameter entry panels, the end of data is the rightmost entry field or selection field choice on the bottommost line of the panel that contains an item on which the cursor may be position. The application defines the beginning of data for other panels.

Chapter 6

Action Bar and Pull-Downs

The ***action bar*** is the panel element at the top of a panel and consists of a list of keyword choices that represent groups of related actions a user may request. The groups of actions appear in a ***pull-down***, usually immediately below the action bar, when the user selects an action bar choice.

The actions typically affect information displayed in the panel body, or, in some way, control the user's dialog with the application.

6.1 Use of the Action Bar and Pull-Downs

An action bar should be used when two or more application actions (in addition to the common dialog actions) are available to the user in a panel. If only one action is available, that action is implied by requesting the enter action; an action bar is not needed.

For example, if a panel lists a ***file*** option, it may provide only one action, ***browse a file***. In this case an action bar is not used because selection of ***file*** can cause only one action, ***browse a file***. If, however, ***file*** has two, or more, possible actions (say ***browse a file*** and ***edit a file***), then an action bar is appropriate.

6.2 Action Bar Location

The action bar is located at the top of a panel. For full-screen panels it is the topmost area of the screen. In a window, it is directly below the window title bar.

6.3 Action Bar Layout

The action bar stretches across the full width of a panel, regardless of the number of items in the action bar. Choices are listed horizontally, on one or more lines, with choices separated by two blanks. The first choice is in the fourth position from the left edge of the panel.

For panels in windows, action bar choices must be reformatted to appear in multiple rows and columns as the size of a window is decreased.

On a color screen, the action bar is separated from the panel body below by a color boundary. When color and emphasis changes are not available on monochrome screens, a solid line is used.

On nonprogrammable terminals, extra space may be needed in front of the first action bar choice to accommodate an entry field.

6.4 Action Bar Contents

The action bar contains choices in the form of single- or multiple-word text. Single-word choices are preferable, with multiple-word text used only when necessary.

The first letter of a choice will be capitalized, with the following exceptions:

- When the choice contains an acronym or abbreviation that is normally capitalized, accepted capitalization is used
- When the choice contains a proper noun that contains a capital letter other than the first letter, the accepted capitalization is used.

Action bar choices will not be numbered.

The last two choices in the action bar are reserved. The last and rightmost choice in an action bar is ***help***. The next to the last choice is ***exit***. The remaining choices in the action bar are defined by the application.

Choices in the action should be ordered by frequency of use, most frequently used options should appear first (e.g. leftmost). The ***exit*** and ***help*** are exempt from this rule.

6.5 Action Bar Mnemonics

Each choice will have a unique mnemonic (a single character representing the choice) that can be used to select the choice.

In an action bar the mnemonic ***X*** must be used for the ***exit*** choice and ***H*** for the ***help*** choice.

Users must have the choice of having mnemonics turned on or turned off. Default is on, except on nonprogrammable terminals. On nonprogrammable terminals, when the user has mnemonics turned on, action bars contain an entry field in which a user may type mnemonics representing action bar choices. The entry field is located in front of the first action bar choice and may be from two to four characters long to accommodate the mnemonics for one action bar choice, and up to three more choices in the pull-down and any succeeding pop-up windows. The length of this field should be compatible with the

number of levels of choices provided to the user. If the example only supports an action bar and pull-down choices, the entry field should be two characters long.

6.6 Action Bar Selection Cursor

The appearance and movement of the selection cursor is covered in "Selection Cursor," section 4.0. The action bar selection cursor covers the entire choice upon which it rests (including the single blank before and after the choice).

On nonprogrammable terminals, the cursor is positioned on the blank space in front of each action bar choice when the user presses the Tab key.

6.7 Action Bar Unavailable Emphasis

Action bar choices are always available so the unavailable emphasis should not be used in an action bar.

6.8 Action Bar Selected Emphasis

Selected emphasis indicates which action choices a user has selected. A selected action bar choice is highlighted with the selected emphasis while its pull-down is displayed. (See Tables 13.6 to 13.12.)

6.9 Action Bar Pull-down Location

An action bar pull-down will be located under (attached) to the action bar. The first character of a pull-down choice will be directly below the first character of its action bar choice. If this alignment is not possible because the pull-down is wider than the space allowed to display it, the pull-down must be repositioned horizontally to make it fully visible. When a pull-down is visible, its corresponding action bar choice should be displayed with selected emphasis.

6.10 Action Bar Pull-down Layout

Pull-down choices are left-aligned in one or more columns. If numbered, the number is the first character of the choice text.

When available, double-line borders should be used; if not available, single-line borders should be used. Multiple selection fields are separated with extra spaces.

The enter and cancel choices in a pull-down will be separated from other choices by using a separator line.

6.11 Action Bar Pull-down Content

See section 4.0, "Selection Fields," for details on selection fields. Pull-downs can contain only selection fields; entry fields are not allowed in pull-downs, with the following exception. On nonprogrammable terminals, an entry field is placed next to the first choice of single-choice selection fields and an entry filed next to each choice of multiple-choice selection fields.

Pull-downs will not contain scrollable selection fields.

If choices are numbered, the text in "Numbering Choices" in section 4.0 should be followed.

Instructions will not be used inside pull-downs.

The directions for use of mnemonics in pull-downs are covered in "Using Mnemonics in Choices" in section 4.0.

6.12 Ellipses for Pull-down Choices

An ellipsis (...) will follow all choices that result in a pop-up window. There are no spaces between the dots or between the ellipsis and the choice text.

6.13 Pull-down Function Key Assignments

Assigned function keys may be made available as part of the choices in a pull-down. A function key is displayed, right-aligned, on the same line as its related choice.

Assigned keys are always active, even if the pull-down is not displayed. For example, if a pull-down has ***Ctrl+C*** assigned to a choice for performing a ***copy*** function, the ***Ctrl+C*** keys can still activate this copy function after the pull-down is gone and the user has moved on to some other part of the application.

Chapter 7

Panel Body Elements

Two of the most important panel elements, selection fields and entry fields, were covered in 4.0, "Selection Fields and Entry Fields." This chapter will describe other panel body components.

Every panel will have a panel body, which can be divided into several panel body areas. Several panel body areas may be used if the application needs to show the user more than one area of information at a time, or if it has to allow a user to enter or update more than one area of information at a time. Although a panel body can contain more than one panel body area, a user interacts with them one at a time. A user can switch to and from panel body areas.

Most of the panel elements described in this chapter are located within a panel body area. Some of the panel elements are located in area of their own within a panel body area.

The panel body can also contain a message area, in which messages appear, and a command area, in which a user types application or system commands.

Layout characteristics not specified here, or in section 4.0, are left to the application developer.

7.1 Panel Body Area Separators

Panel body area separators distinguish the panel body from adjoining areas, such as the action bar and function key areas. Separators are used so that the user knows where panel parts begin and end (helpful for knowing where things like cursor-driven scrolling will take place).

7.1.1 Body Area Separator Content

A panel body separator can be formed with a color boundary, solid line, blank line, column headings, message area, or command area.

7.1.2 Body Area Separator Requirements

A color boundary separator will be placed between the action bar and the panel body. If color is not available, a solid line will be used.

A color boundary separator will be placed between the panel body areas that a user can switch to. If color is not available, a solid line will be used.

7.2 Panel ID

A panel ID is an element of the panel body consisting of alphanumeric protected information that uniquely identifies the panel and gives the user an easy way of identifying the current position within a dialog. It is a user option as to whether the panel IDs are visible or not.

7.2.1 Panel ID Location

The panel ID is located on the same line as the panel title.

7.2.2 Panel ID Layout

The ID will be left-justified, keeping some separation between it and the panel title. If there is not enough room on the line for the title and ID, the ID will be truncated (from the right).

7.2.3 Panel ID Contents

The panel ID is a unique, alphanumeric character-string identifier within the application containing no blanks. The ID will be 10 characters, or fewer, in length.

7.3 Panel Title

A panel title is the element that tells a user what the information in a panel body area is about. All panels must have a panel title.

Messages (pop-up window) do not have panel titles, since the message should be self-explanatory.

If the panel is in a window that has a window title bar, you can use the window title in place of the panel title if the window will contain the same panel for the duration of its display.

Panel titles are located at the top of the panel body area.

7.3.1 Panel Title Layout

The panel title will be centered and will stay centered if the size of the window changes. The title is truncated from the right when the panel body area becomes too small.

The title will not be part of scrollable area (title will always be visible).

The title should be followed by a blank line (which may be removed if the window becomes smaller).

7.3.2 Panel Title Content

The panel title is protected (cannot be changed by a user). The title may contain variable information (file names, etc.), but cannot contain entry or selection fields. Normal capitalization should be used.

The title or panel instruction(s) should identify any default action.

7.4 Instructions

Instructions are panel body elements that tell a user how to interact with a panel and how to continue with the application. There are two locations for instructions: the top of the panel body (below title and/or action bar) and the bottom of the panel body (above function key area). Instructions at the top are preferred, but sometimes instructions need to be at the bottom.

Instructions should be used in most panels.

Instructions should be included to tell users how to switch from the panel body to the action bar (if appropriate).

7.4.1 Instruction Layout

Instructions are one or more lines of text in mixed case and left-justified in the panel body. Instructions are truncated from the right when a window displaying the panel becomes too small to display all the instructions.

7.4.2 Guidelines for Writing Instructions

The verb ***type*** (e.g., ***Type Name...***) should be used, rather than the verb ***key*** (e.g., ***Key in Name...***).

The action ***select*** should be distinguished from the action ***enter***. ***Select*** means to mark a choice, whereas ***enter*** means to send the panel to the computer for processing.

The verb ***press*** should be used to tell a user to initiate an action through the keyboard (e.g., ***Press F1 for Help***).

The verb ***request*** should be used to tell a user to initiate an action with a keyboard OR mouse (e.g., ***Request Prompt for Help List***).

7.5 Headings

Headings are elements of the panel body that identify columns and groups of related items.

7.5.1 Column Headings

Column headings should be used to identify entry fields or selection fields when all items in the column are of the same type.

Every field should have some sort of identifier (column heading, field prompt, title) unless it is the only field in the panel and the panel title explains the field.

Column Heading Location

Column headings are located above the fields or columns of fields they identify. Column headings will be in a separate area that does not scroll if the information can be scrolled vertically (the heading must be kept visible).

Column Heading Layout

Column headings can use mixed case and should follow normal rules for capitalization. A column of information is as wide as its longest item of information, or its longest column heading. Headings should be centered over the appropriate columns.

7.5.2 Group Headings

Group headings will be used to identify a group of entry fields or selection fields. Group headings might be used concurrently with field prompts when the group heading identifies a group of fields and the field prompts identify the individual fields.

Group Heading Location

A group heading should be placed above each group of fields. Indent the field prompts two positions under the group headings. If a group heading is located in a panel body that scrolls, the heading will also scroll.

Group Heading Layout

Group headings will be left-aligned in a panel area. Indent group headings two positions following instructions. Group headings will use mixed case and follow normal rules for capitalization.

7.6 Field Prompts

Field prompts are elements of the panel body that identify selection fields, entry fields, and variable information. The user cannot select or change a field prompt.

A field should use a field prompt, or other identifier, for each field unless the field is the only field in the panel and the title explains the field.

7.6.1 Field Prompt Location

Field prompts are located to the left of the fields they identify.

7.6.2 Field Prompt Layout

Field prompts are left aligned in the panel body, using mixed case. If a field prompt falls under a group heading or instruction, it should be indented two spaces.

Leader dots (periods) should be used to connect field prompts and fields so that the user can easily read the panel. There should be a minimum of two dots on a line when leader dots are used. Leader dots are not needed for the longest field prompt in a group of aligned fields, or for a panel with only one field. A space should be left between each dot and three spaces between the last dot and the field.

There should always be at least one space between the field prompt text and the first leader dot.

Field prompts for protected information (unchangeable by user) should always be followed by a string of leader dots, with the last dot being replaced with a colon (:) to indicate that the adjacent field is protected.

7.7 Scrolling Information

Scrolling information tells a user the relative positions in a file, that more information exists outside the visible area, and in which direction to scroll the panel body to reveal that information.

7.7.1 Text Scrolling Location Information

Text scrolling information is optional, it is text with numbers that tell a user the position of the visible information relative to additional, nonvisible information (example: "Lines 1 to 15 of 200").

As an application option, text scrolling information may be included in each panel body area that scrolls.

7.7.2 Text Scrolling Location Information Location

Text scrolling information will be right-justified on a line above the panel body area. It can appear on the same line as the panel title or top instructions. If used with scrolling arrows, it should appear above the arrows.

7.7.3 Text Scrolling Location Information Layout

The text scrolling information should remain right-justified as a window is resized. Keep space between this information and any other items on the same line. If the line needs to be truncated, the scrolling information should be truncated first (from the right).

7.7.4 Text Scrolling Location Information Content

The format for text scrolling information is:

Aaaaaaaaaaaaa xx to yy of zz

where:

Aaaaaaaaaaa is the appropriate term for the information itemsbeing displayed (e.g., ***lines***, ***entries***, ***parts***, etc.). The application supplies the appropriate term.

xx to yy is the range of items being displayed.

zz is the total number of information items in the list.

Text scrolling location information can be used to indicate the vertical and/or horizontal ranges.

7.7.5 How a User Interacts with Text Scrolling Location Information

A user may be allowed (application option) to type over the first value (the *xx*) of the text scrolling location information to reposition the list of items relative to the new starting point (*xx* is an entry field). Only valid values can be accepted; if the value entered by the user exceeds the range in either direction, the list is repositioned to the top or bottom (in the direction indicated by user).

A mouse cannot be used on the text scrolling location information. Scrolling arrows must be used for that purpose.

7.7.6 Scrolling Arrows

Scrolling arrows indicate that additional information exits outside the visible panel body, and show in which direction to scroll and which arrow to use to reveal that information. Each panel body area in a panel body that scrolls has its own scrolling arrows.

7.7.6.1 Scrolling Arrows Location

Scrolling arrows will be right-justified on a line above the panel body area to which it applies and below text scrolling information (if used).

7.7.6.2 Scrolling Arrows Content

Scrolling arrows contain the word ***more:*** followed by the appropriate arrows. Example: ***more:*** ← ↑ ↓ →.

Scrolling arrows are all on one line with one space between the colon and the left arrow and between the rest of the arrows.

An arrow is only displayed if there is more information in the direction it points. A user then presses the corresponding key to scroll the information into view.

On nonprogrammable terminals the arrows are <—+ >.

7.7.6.3 Scrolling Arrows Layout

Scrolling arrows remain right-justified with the margin of the panel body area as the window is resized. Spaces should be kept between the arrows and any other items on the same line. If the window is reduced in size and truncation is required, truncation on the scrolling arrows will be from the right.

The scrolling arrows should be laid out with one space between the colon (:) and the first arrow position and one space between the arrows.

The entire space should be reserved for scrolling arrows (the word ***more:*** and the four arrows) even if on the current display no scrolling is available (all information is visible). There should be consistency: always have the left arrow position first, then the up arrow position, then the down arrow position, and then the right arrow position.

7.7.6.4 How a Keyboard User Interacts with the Scrolling Arrows

A keyboard user presses the indicated arrow keys (or symbols) to display more information in the direction of the arrow. If the panel area contains a cursor, the panel area scrolls when the cursor reaches the panel area boundary. If the panel area does not contain a cursor, scrolling occurs immediately.

7.7.6.5 How a Mouse User Interacts with the Scrolling Arrows

A user can cause scrolling by placing the mouse pointer on a scrolling arrow (or symbol) and pressing the select button. Continuous scrolling results when the button is held down.

7.8 Message Area

It is recommended that messages be presented in pop-up windows. However, an area (one or more lines) may be set aside near the bottom of the panel body to display messages.

A message area should be used when it is important not to interfere with the information in the panel (e.g., when it is desirable to not overlay information with messages).

A message area might also be used when a user is not required to respond to the message. If a user response to a message is required, a pop-up window is recommended.

Use a message area if a message is a notification or warning message. Critical messages should be displayed in a pop-up window.

A message area is useful for displaying messages that disappear upon the next user action or some other specified action.

7.8.1 Message Area Location

The message area is to be located above the command area, if present, at the very bottom of the panel body. The message area can also act as a separator between a panel body area and the command area or function key area (if either is used).

7.8.2 Message Area Layout

Messages are left-justified within the message area.

Chapter 8

Command Area

A command area provides a user the capability to request actions by entering commands. For experienced users this can provide a "fast-path" in using the application. The command area can be used as an alternative to the action bar and pull-downs. The command area may be located in a base panel or in secondary panels.

8.1 Command Area Location

The command area can be in any of the following locations:

Panel Body Area. The command area is at the bottom of the panel body, just above the function key area (if used). Using this approach the command area will not obscure the panel body. It is also used when secondary windows are not available. Though less desirable, a command are can be located at the top of the panel body, below the action bar (if used).

Pop-up Window. This is useful when secondary windows are not available.

Secondary Window. This is the preferred approach, a secondary window for input of commands. It can be used by the user if it obscures information on the primary window. This window is accessible from other windows.

8.2 Command Area Content

All command areas, regardless of how they are displayed, will contain a field prompt and an entry field. The entry field may be scrollable except when the window it is in is at its maximum size. The field prompt consists of the word **command**, a blank space, and a right-pointing arrow (three equal signs and the greater-than symbol). No leader dots are used.

If the command area is scrollable, use the normal delimiters for scrollable fields (underscore character). Also use the greater-than and less-than symbols to indicate the direction the field scrolls.

A command area in a pop-up window will contain, at a minimum, a function key area and a panel body in which the command entry field is located. The panel body might also contain some instructions.

8.3 The Command Dialog Action

If the command area is located in a secondary or pop-up window, the window appears when the user requests the command dialog action by pressing the key assigned to the action, or by selecting a pull-down choice. If the user selects the command action from a pull-down, the pull-down disappears when the command window appears.

The user can exit the secondary window, or cancel the pop-up window, when the command area is no longer needed. A user cannot switch from the pop-up window to the underlying panel.

When the command area in the panel body is not visible, a user requests the command dialog action to make it appear. It stays visible until the user requests the command action, while the cursor is in the command area, unless the user has decided to keep it displayed all the time. Each time the command area is turned on, the cursor moves to it. When the command area is turned off, the cursor returns to its original position in the panel.

When a command area in the panel body is visible, a user may request the command action to move the cursor back and forth between the command area and the panel body.

A user may also use the switch-forward action to move the cursor to and from a command area in the panel body area when the command area is visible. A user switches forward to move the cursor sequentially through other panel areas until it reaches the command area.

On nonprogrammable terminals, the arrow, Tab, and New Line keys may move the cursor to and from the command area.

When the cursor is in the command area entry field, a user types in the command and then requests ***enter***.

8.4 The Retrieve Dialog Action

A user requests the retrieve dialog action while a command area is visible to redisplay the last command that was issued. The previous command appears in the entry field. A user may then change the command, add parameters, or reenter the command. If the user requests a retrieve while the command area is not visible, it has the same effect as the command action and also redisplays the last command.

The user may request retrieve repeatedly to back up through the list of previous commands in last-in-first-out order. The application determines how many commands the retrieve action saves.

8.5 The Prompt Action

As an application option, a pop-up window containing a list of valid commands appears when a user requests the prompt dialog action while the cursor is in a blank command area entry field. A user can select a command from this list.

If the user types a command, or part of a command, in the entry field and then requests the prompt action, a panel in a pop-up window appears to help the user complete the command parameters. The pop-up should contain as much of the command as the user typed and should prompt the user for the remaining command parameters.

If an action is available from both the action bar and by entering a command in the command area, the panels that prompt users to complete the action are the same in both cases.

8.6 Command Validity Checking

The application (for application *commands*) or the system (for system commands) must determine the validity of commands. If the user enters a command that is invalid, a message appears.

8.7 Using a Command Area and the Action Bar Together

If an application uses an action bar and a command area, at the same time, the two interface elements must be consistent. When functions are available through both the action bar and the command area, the command area is considered the "fast-path" way of accessing the action bar functions.

The action bar might not contain all commands available to a user. Those commands that are available from the action bar and through the command area must have the same name in both places. The same name cannot be used for different functions.

Usually an action will be in the pull-down, and its parameters will be in subsequent pop-up windows. In some cases, an action bar pull-down choice will refer to a group of actions, rather than a single action, and the resulting pop-up window will list the specific actions. In this case the choice in the pop-up window should be the same as in the command area.

Regardless of whether a user selects the action from a pull-down or pop-up window, the parameters for that action should be specified in one or more pop-up windows. These parameters should correspond to equivalent command language parameters.

The command parameters available from the action bar pull-downs and subsequent pop-up windows might be a subset of all parameters that a user can use in the command area.

Chapter 9

Function Key Area

The ***function key area*** is the panel area at the bottom of a panel that lists available actions and their key assignments. Some actions are common to all applications while others are application specific.

The actions that appear in the function key area at any point are available for the current state of the application. If an action is not currently listed in the area, then it is not available to the user. When a user selects an action from the key area, the action occurs immediately.

9.1 Using the Function Key Area

The actions in the function key area represent the minimum set of actions required in a dialog of more than a few panels. The simplest panels and dialogues often will use only a function key area and no action bar, even if the only action available is ***cancel***. Panels displayed in pop-up windows typically contain only a function key area and no action bar, because the dialog in pop-up windows is often simple and only requires a user to provide information.

Each panel will have a function key area defined. The user must be given the option of specifying if, or how, the function key area is displayed.

9.2 Three Forms of Function Key Area

A user may display the function key area in any one of three forms by requesting the display keys actions. Whichever form a user selects, it must be propagated to all windows and panels for the application.

The three forms are:

Long. This displays the actions that appear in the short form along with a set of actions that are determined to be needed for that panel. These actions may be all or part of the actions available on the panel. If an action bar is present, some actions may be duplicated. Duplicate actions will be assigned the same keys in both the function key area and the action bar pull-downs.

Short. In this form, only some of the dialog actions are displayed (those that a mouse user might otherwise have to select from the keyboard). All key

assignments are still in effect. Putting a short list of actions in the function key area means a mouse user will not have to switch to the keyboard. For example, display the enter action so that a mouse user can select it from the short-form key area rather than having to press the ***enter*** key. Actions available on the action bar should not be included (except for ***help*** and ***exit***).

No Display. This removes the function key area, making the space available to the application. The key assignments are still in effect, they just are not displayed. Since they are not displayed, they cannot be selected with a mouse. The choices might still be available as pull-down choices.

9.3 Function Key Area Location

The function key area will be located at the bottom of the panel.

9.4 Function Key Area Content

Only actions valid for the current panel will be displayed. No space is reserved in the function key area for unused keys.

For each dialog action available in the current panel a key should be assigned so that the user might request the action by simply pressing the associated key.

9.5 When to Use Dialog Actions in the Short-Form Function Key Area

These dialog actions must be displayed, when they are available, in the short-form of the function key area. They are: ***Cancel***, ***Enter***, ***Exit***, ***Help***, ***Prompt***, ***Refresh***.

For help panels, the following actions must be displayed: ***Extended Help***, ***Help Index***, ***Keys Help***.

Cancel

This gives the user a chance to back up from one panel to the previous panel. Cancel must be included in all pull-downs. Cancel should be included on panels inside pop-up windows, with the exception of panels used as notification messages when they contain an ***enter to continue***.

Cancel may be included in other panels as an application option. It is recommended that cancel be in all panels that are part of a sequence of panels which comprise a single unit of work.

If cancel is assigned to several keys (for example, Esc and F3), a user can request cancel with either key. Use of dual keys for an action is discouraged.

Enter

This should be included if the panel contains any entry fields or if users can select more than one selection-field choice from the panel. This includes panels that contain a multiple-choice selection field, and those that contain more than one single-choice selection field.

For a mouse user, the ***Enter*** provides a "touch area" for the user to enter the panel.

Exit

When a user is allowed to terminate the current function from the current panel, an Exit must be included. The Exit must be available from the primary window of an application in the action bar, if used, and the function key area.

If the application supports several levels of exit, the lowest level exit is to be displayed in the short-form function key area.

Exit should be used in pop-ups only if it is a help pop-up. Cancel should be used in other types of pop-up windows. In other cases, the exit and cancel actions may appear in the same panel.

Help

Every panel should have a help action. The help action always appears in an action bar, if used. The help action appears in the function key area of panels that do not have an action bar.

Help depends on the position of the cursor to indicate the item for which a user is requesting help.

Extended help. This provides a user information about an entire application panel rather than a particular item in a panel. Extended is required within help panels, except in the extended help panel themselves.

Help Contents. This displays a table of contents, organized by topic, of the help information available to a user.

Help Index. This provides the user with an alphabetic index of help information. A help index is required in help panels, except for the help index panel itself.

Keys Help. This provides a user with the names and functions for all the keys used by the application. Keys help is required in help panels.

Tutorial Help. This lets a user access an application's tutorial; it is optional.

Prompt

The prompt action will be included when a panel contains an entry field that the application can fill in, based on what a user selects from a list of entries. When an application knows the possible entries for an entry field, a prompt action should be included so that a user can recognize and select the desired choice rather than having to remember all the choices and typing the desired one.

Prompt depends on the position of the cursor to indicate the item for which a user is requesting a prompt.

Refresh

For an entry panel, refresh restores the panel to its initial state, discarding any information a user may have typed in.

For a panel which displays the current status of objects, the refresh action redisplays the panel with any updates to those objects. If this panel also allows a user to type information, Refresh discards any information typed in by the user.

Refresh is optional, though recommended for panels containing selection or entry fields.

9.6 Long-Form Key Area Contents

The application determines the content of the long-form function key area. The minimum content is the applicable dialog actions that appear in the short-form plus:

Backward, ***Forward***, and ***Switch*** to action bar (when these apply).

Other actions frequently used by users, or that are common to many panels, should be assigned to the long-form function key area. Actions that rely on the location of the cursor are particularly suited for the function key area.

9.7 Function Key Action Abbreviations

The following abbreviations are allowed for function key assignments:

Action	*Abbreviation*
Backward	Bkwd
Display keys	Keys
Extended help	Ex help
Forward	Fwd
Switch to action bar	Actions
Switch backward	Switch bkwd
Switch forward	Switch fwd

No other actions may be abbreviated.

9.8 Function Key Area Layout

The function key area stretches across the full width of the panel. If the panel is in a window, the function key area is not reformatted when the window is sized smaller. The area is truncated if the window is too small to display all of the actions.

Actions are listed horizontally, with the first action choice indented one position from the left edge of the panel. In both short and long forms, the function key area can take up several rows. If more than one row is used, the choices should be aligned in columns.

When a user changes the form of the function key area and more rows are needed, the panel body may be compressed or the window may be expanded vertically to provide additional space.

The function key area is always a separate panel area and a user must use ***switch forward*** and ***switch backward*** to access it (nonprogrammable terminal users can also use the arrow keys).

The function key area may be separated from the panel body with a solid line, color boundary, or a blank line. The separation is not required.

Once the cursor is in the function key area, a user may place the selection cursor on an action and select it.

9.9 Order of Actions

The dialog actions must be laid out in a certain order, depending on whether the long or short form of the function key area is displayed and what type of panel they appear in. Only available actions are to be listed.

9.10 Short Form Function Key Area

This is the order of dialog actions in short-form function key areas.

Application Panels:

Enter Esc=Cancel F1=Help F3=Exit F4=Prompt F5=Refresh

For Application panels on nonprogrammable terminals:

Enter F1=Help F3=Exit F4=Prompt F5=Refresh F12=Cancel

Help Panels:

Enter Esc=Cancel F1=Help F2=Extended Help F3=Exit
F4=Prompt F9=Keys help F12=Help index

For Help panels on nonprogrammable terminals:

Enter F1=Help F2=Extended help F3=Exit F4=Prompt
F9=Keys help F11=Help index F12=Cancel

9.11 Long Form Function Key Area

This is the order of dialog functions in long-form function key areas.

Application Panels:

Enter Esc=Cancel F1=Help F3=Exit F4=Prompt F5=Refresh
F6=Switch fwd F7=Backward F8=Forward F9=Retrieve
F10=Actions

For Application panels on nonprogrammable terminals:

Enter F1=Help F3=Exit F4=Prompt F5=Refresh F6=Switch fwd
F7=Backward F8=Forward F9=Retrieve F10=Actions F12=Cancel

Help Panels:

Enter Esc=Cancel F1=Help F2=Extended help F3=Exit F4=Prompt
F5=Refresh F7=Backward F8=Forward F9=Keys help
F10=Actions F11=Help index

For Help panels on nonprogrammable terminals:

Enter F1=Help F2=Extended help F3=Exit F4=Prompt F5=Refresh
F7=Backward F8=Forward F9=Keys help F10=Actions
F11=Help index F12=Cancel

9.12 Function Key Area Emphasis

When a user interacts with a panel, the action choices in the function key area are available and displayed with normal color and emphasis as described in Tables 13.6 to 13.12. The unavailable emphasis is not used in the function key area.

9.13 Pointed-At Emphasis

When a user points at an action with a mouse pointer, while pressing the select button, the action is emphasized the same way a selectable choice in a panel body is emphasized when the cursor is on it. (See "Selection Field Selected Emphasis.") The selection cursor does not move from the panel body to the function key area in this situation.

9.14 Selected Emphasis

When a user selects an action in the function key area, the action is emphasized the same way a selectable choice in a panel body is emphasized when it is selected, except that no selection indicator appears next to it. Because all actions occur immediately, a user may see only a brief color change before the action begins (nonprogrammable terminals will display the selected emphasis until the action is completed).

9.15 How Users Interact with the Function Key Area

A user requests an action in the function key area by pressing on of the assigned keys or by using the point-and-select technique with the selection cursor or mouse pointer.

9.16 User Interaction Using a Keyboard

With the point-and select approach, a user moves the selection cursor to the function key area using the switch forward and switch backward actions and presses the arrow keys to move to the desired action. When the selection cursor is on the desired action, the user requests an enter. On nonprogrammable terminals, the arrow keys will move the cursor one position at a time in the function key area; the Tab key is not supported in the function key area.

9.17 User Interaction Using a Mouse

A mouse user using the point-and-select approach will move the mouse pointer to an action and press the select button. The action is highlighted. The selection cursor does not move to the function key area in this case. When the user releases the select button, the action is emphasized with selected emphasis, and the action executes.

Chapter 10

Scrolling Panel Body Areas

A user may scroll an entire panel or areas of a panel using two scrolling methods: cursor-driven scrolling and cursor-independent scrolling. This section explains these two methods.

A user can scroll only the panel body that is currently interactive. If the user wishes to scroll an area that is not currently interactive, the user must first make that area interactive (e.g., switch). The interactive panel body area is the one in which the cursor is located.

A panel body area stops scrolling at the boundary of the information. Scrolling arrows, text location information, etc. is explained in "Scrolling Information". A visual indicator may also be placed at the boundaries of the information (e.g. "End of Data"). When the bottom or top of information is reached, the application must inhibit any further scrolling in that direction.

Information that scrolls does not wrap.

10.1 Cursor-Driven Scrolling of Panel Body Areas

A user scrolls a panel body area by moving the selection cursor or text cursor until the cursor encounters a panel body area boundary. The panel body area then scrolls the same increment as the cursor moves (provided there is information to be scrolled in that direction). Nonprogrammable terminals cannot make use of cursor-driven scrolling.

10.1.1 Cursor-Driven Scrolling with a Keyboard

The following keys move the cursor and can cause a panel to scroll:

- The Left, Right, Up, and Down arrow keys
- The Tab and Backtab keys
- The Beginning-of-Data and End-of-Data keys
- The Beginning-of-Line and End-of-Line keys
- The Beginning-of-Field and End-of-Field keys

10.1.2 Cursor-Driven Scrolling with a Mouse

A mouse user can scroll a panel by pressing and holding the select button, then moving the selection cursor with the mouse pointer. When the cursor is at a boundary, the user continues to scroll the information by pressing and holding the select button.

10.2 Scrolling in Truncated Fields

A panel in a window that is resized smaller than full size may have its entry fields or selection fields truncated. If the user moves the cursor in either an entry field or selection field, the panel body area scrolls until the truncated portion of the field is visible. Then the information in the field scrolls.

10.3 Cursor-Independent Scrolling of Panel Body Areas

With cursor-independent scrolling, a user can scroll a panel in up to four directions when information is beyond the boundaries of the visible area. Cursor-independent scrolling is performed only if there is additional information to be seen. Scrolling must result in a valid cursor position.

The scrolling actions for cursor-independent scrolling are:

- ***Backward***—displays information above that currently visible in the panel body area.
- ***Forward***—displays information below that currently visible in the panel body area.
- Left—displays information to the left of that currently visible in the panel body area.
- ***Right***—displays information to the right of that currently visible in the panel body area.

As an application option, cursor-independent scrolling of panel body areas keeps the cursor stationary on the screen, or in the information. If the cursor remains stationary in the information being scrolled, it is an application option to either stop the cursor at the boundary of the panel body area, or move it to the first visible field in the panel area when the choice it was on scrolls out of view.

10.4 Cursor-Independent Scrolling Increments

There are three classes of scrolling increments for the scrolling actions. The user can choose one of these classes by selecting an action bar pull-down choice that results in a pop-up window, or by typing a command in a command area.

Panel-Area-Related Increment. The visible panel-area-related increment varies with the window size. The user sees all the information as it scrolls and no information is skipped. The panel body area-related increments are:

- The full, visible, panel body area
- The visible panel body area minus one item (for forward and backward scrolling)
- A portion, such as half, of the visible panel body area
- One-third of the visible panel body area for left and right scrolling

The default increment for backward and forward scrolling is the visible panel body area minus one item. The full, visible, panel body area may also be made the default. The default for left and right scrolling is one-third of the visible panel body area.

Fixed Increment. This allows a user to scroll information a fixed number of lines. Depending on the size of the window, some information may be skipped while scrolling.

Application-dependent Increment—allows a user to scroll information any amount that is meaningful to the application.

10.5 Scroll-Lock Mode Option

The scroll-lock mode is an application option that lets a user change the arrow keys from keys that move the cursor to keys that scroll the information. Scroll-lock mode provides a similar effect with a mouse.

The scroll-lock mode option is not available on all nonprogrammable terminals.

A keyboard user goes into scroll-lock mode when the user presses the Scroll Lock key, which is an on-off toggle key.

If scroll lock is provided for a mouse, the mouse user presses the Scroll Lock button (usually button 2). When this is pressed, the interactive panel body area switches to the panel area containing the mouse pointer. Then, while the user holds the button down and moves the mouse, information moves in the same direction.

When the button is pressed, the mouse pointer changes appearance to let a user know the scroll-lock mode is on (the reverse color up/down arrow).

Scroll lock is a keyboard state. If an application supports the scroll-lock mode option, it must switch into that mode when a user requests it from within the application, or while in a different application and switch back to the application that supports the option.

10.6 Scroll-Lock Mode Option Scrolling Increment

The scrolling increment when a user scrolls in scroll-lock mode is the same as for cursor movement, a line at a time.

10.7 Coordinated Scrolling Panel Body Areas

Panels may be created to allow a user to scroll several areas of a panel at the same time. If such areas contain information that is related to information in other panel areas, the two areas must maintain their relationship when the user scrolls one of the areas. This is called coordinated scrolling and is possible horizontally and vertically.

Chapter 11

Switching Between Areas

This section describes the switching actions that move the selection cursor to another area in the panel.

11.1 The Switch Forward and Switch Backward Actions

The switch forward and switch backward actions are interaction techniques that allow users to transfer the selection cursor from one area of the panel to another.

The switch forward action forwards the cursor from area to area in the left-to-right and top-to-bottom direction. The switch backward action moves the cursor from area to area in the right-to-left and bottom-to-top direction.

The switch actions are required when there is more than one panel body area that cane be interactive. A user can switch forward and backward to any panel body area that contains selection fields or entry fields, or that a user can scroll. The command area and function key area are panel areas that the user can switch to. The action bar is treated differently; a user switches to the action bar with the switch-to-action-bar action.

The switch actions transfer the keyboard input from one area to another within the same window.

The switch backward is an application option. The more areas in a panel, the more beneficial switch backward becomes.

11.2 How A Keyboard User Switches Areas

The keyboard user switches the interactive cursor from one area to another by pressing the Switch Forward key. This key does not move the selection cursor to the action bar; this is a different action using a different key. The Switch Forward key does move the cursor to the panel body command area and the function key area if they are visible.

The switch forward action skips over areas that cannot be interactive.

11.3 How a Mouse User Switches Areas

A mouse user switches areas by placing the mouse pointer in the desired area and pressing either the select button or the scroll-lock button (if supported). The switch occurs when the button is pressed. A user can select a choice by pressing and releasing the button.

11.4 Switching to the Command Area

A user may enter commands in a command area that is in either a pop-up window, secondary window, or an area of the panel body. If the command area is in a panel body, the following actions apply:

- If the command area is visible, a user may request to switch forward and switch backward to move the cursor through other panel areas to the command area. The command action moves the cursor directly to the command area from any panel body area and returns it to the panel body area it came from.

- If the command area is not visible, a user must request the action to make the command area to appear, the cursor then moves to it. When the user requests the command area to disappear, the cursor returns to its original position in the panel.

11.5 Switching to the Action Bar

One way a user may switch from the panel body to the action bar is by pressing the Switch-to-Action-Bar key. The selection cursor appears on the left-most choice in the action bar. The pull-down for that choice does not appear. See "Switching the Selection Cursor to the Action Bar Using a Keyboard" and "Switching to the Action Bar Using a Mouse" for information about other ways a user can move the cursor to the action bar.

On nonprogrammable terminals a request to switch-to-action-bar will cause a host interrupt (the system knows where the cursor was positioned). When the user switches back, the cursor can return to its original position. A user can also move to an action bar by using the arrow and Tab keys (and Home key on some). None of these keys cause a host interrupt, so the system cannot return the cursor to its original position.

Chapter 12

Panel Types

This section will discuss five different types of panels:

Menu Panels, Entry Panels, Information Panels, List Panels, Logo Panels.

By combining these panels, "mixed" panels may be created. Limiting the number of different types of panels will make it easier for a user to learn an application or system.

12.1 Menu Panels

A menu panel displays one or more lists of choices from which a user may select one or more choices.

12.2 Panel Elements in Menu Panels

Action Bar

Some menu panels may need an action bar, others may not. An action bar should not be used if a user can request only one action to work on the choices in the panel. For example, a menu panel with the title ***View Accounts*** may offer the user only one action, ***view***, from several different types of ***accounts***. An action bar is not needed because the user is offered only one action. If an action bar is used, it is used in the normal way; user selects one or more from the panel body and then select an action from the action bar.

Instructions

Single-field panels should contain instructions to indicate how many choices a user can select. Panels with several fields should give specific instructions about the number of choices a user can make in the panel. Specific words such as ***tasks, objects, parts, orders, customers***, etc., should be used to identify the choices to the user.

Panel Body Areas

A menu panel contains a single panel body area that does not scroll or one panel body area that does scroll for the choices, and one that doesn't scroll for protected information such as panel title, instructions, and headings.

Selection Fields

This includes one or more selection fields and appropriate field prompts and headings. Selection fields should be arranged in rows and columns.

Entry Fields

There are not to be used in menu panels.

Field Prompts

These are used when a group of selection fields is displayed vertically in a panel body area (group headings might also be used). In single-field panels, a field prompt is not required because the panel title should identify the field.

Column Headings

These are used when more than one field is displayed in several columns. In a single-field panels, a column heading is not required since the panel title should identify the field.

Selection Cursor

The selection cursor should initially be located on the topmost and leftmost field in the panel body. The initial selection cursor position is within the first field as defined in "Selection Cursor Initial Position."

12.3 Entry Panels

Entry panels let a user type in entry fields and select choices from selection fields. The information entered in an entry field may be information entered into a file, parameters, or options associated with an action request that the computer needs so it can complete processing.

There are three styles of entry panels:

Parameter Entry - used for telling the system parameters and options associated with an action request.

Form Fill-in Entry - used to record information that is to be filed. This style may look similar to a paper form.

Tabular Entry - used to record information that is to be filed. This style contains entry fields arranged in rows and columns that are identified with headings.

12.3.1 Panel Elements in All Entry Panels

This section lists the rules for panel elements associated with all three styles of entry panels. Other rules for use of panel elements are elsewhere in this document.

Panel Body areas - one panel body area that does not scroll should be used, or one scrollable panel body area for the entry and selection fields and one panel body area that doesn't scroll for panel titles and instructions.

Entry Fields - one or more must be included. Each may or may not contain initial values. Protected descriptive text may accompany entry fields. If prompts are used, descriptive text should be used to let the user know that a prompt is available.

If an entry fields accepts a certain keyword word, descriptive text should inform the user of the keyword. If more than one entry field appears on a line, at least four spaces will separate the end of one field from the beginning of the next field prompt.

Required entry fields will be placed at the top of the panel body area, followed by optional entry fields, in order of importance, unless some other rule takes precedence.

Selection Fields Optional.

12.4 Parameter Entry Panels

The parameter entry panel displays fields that let users type in parameters or specify options for actions requested. Parameter entry panels contain entry fields and may contain selection fields. Parameter entry panels should be used for action requests and simple information-entry situations.

12.4.1 Panel Elements in Parameter Entry Panels

This section covers rules for panel elements associated with parameter entry panels.

Action Bar

Usually not used with parameter entry panels because user is responding to a requested action. An action bar might be included if other actions need to be available.

Panel Body Area

See "Panel Elements in All Entry Panels." All fields will be arranged in a single column. When there are too many potential selections for a selection field, a scrollable entry field or an entry field.

Headings

No column headings will be used. Group headings are optional.

Field Prompts

Field prompts are required, except for single-field panels (the title will identify the single-field).

12.5 Form Fill-In Entry Panels

Form fill-in entry panels typically contain entry fields arranged to resemble paper forms.

12.5.1 Panel Elements in Form Fill-In Entry Panels

This section will discuss rules for panel elements associated with fill-in entry panels.

Panel Body Areas

See "Panel Elements in All Entry Panels." One or more entry fields will be included. Selection fields may also be included. Attempts should be made to align field prompts, fields, and descriptive text for entry fields in columns to make the form easier to use.

Headings

Optional

Field Prompts

Required except for single-field panels (title describes the field)

Tabular Entry Fields

Tabular entry panels have one or more columns of entry fields. Panel Elements in Tabular Entry Panels

This section covers rules pertaining to use of panel elements in tabular entry panels.

Panel Body Areas

See "Panel Elements in All Entry Panels."

Headings

Column headings are required.

Entry Fields

All entry fields within a column should be aligned. The leftmost column will be indented two spaces under the instructions.

12.6 Information Panels

The information panel displays protected information such as messages and help.

12.6.1 Panel Elements in Information Panels

This section covers rules for panel elements associated with information panels.

Action Bars

Normally action bars are not used, but may be useful for some applications.

Panel Body Areas

At least one panel body area must be included. The panel body area contains only protected information, no entry or selection fields. To make text easier to read, right justification and variable spacing will not be used.

12.7 List Panels

A list panel displays a list of choices from which a user may select one or more choices and then specify one or more actions to work on the choices. Only one action at a time works on each choice, but, different actions can work on different choices. For example, in an application that keeps track of documents, a user might use a list panel to select two document names and specify an archive action for one and an erase action for another. Both actions take place when the user requests ***enter***. List panels use a combination of selection fields and entry fields, The list of choices is a multiple-choice selection field. Preceding, or following, each choice is an action entry field in which a user may specify the action to work on that choice.

12.7.1 List Panel Compared to a Menu Panel

The main difference between a menu panel and a list panel is in the number of actions that a user can request. A menu panel allows a user to request one action at a time; the action might apply to several items selected in the panel. A list panel allows a user to request several actions for several choices. The menu panel is simpler, and the list panel is more powerful.

12.7.2 Panel Elements in List Panels

This section discusses actions and instructions for panel elements associated with list panels.

Action Bar

This is required.

Instructions

Instructions are used to tell a user to select choices and actions.

Action Codes

As an application option, instructions may include a list of available actions (and codes representing the actions), as well as information about using these actions and codes. All action codes must also be available from the action bar. These lines of actions and action codes may be turned off and on by the user by selecting a pull-down choice or entering a command. The default should be to not display the actions and action codes.

The actions and action codes will be placed between the instruction line describing their use and the column headings. The actions and action codes will be indented two to four spaces. A mnemonic or number will be assigned to each action. The action mnemonics are emphasized and are active when users have mnemonics turned on; they are inactive when users have mnemonics turned off. The numbers are displayed and are active when a user has numbers turned on; they are inactive and not displayed when numbers are turned off.

One or more lines are used to display the actions and action codes from left to right and then top to bottom. If more than one line is used, the codes should be aligned vertically.

Panel Body Areas

The list panel contains one panel body area that does not scroll or one panel body area that does not scroll for the choices and entry fields and one that doesn't scroll for the panel title, instructions, and action codes.

Column Headings

These are required. The heading above the action entry fields should be "Action" or "Option" (or abbreviate "Act." and "Opt."). Two to seven spaces should be left between the column heading for the action entry field and a preceding column heading.

Entry Fields

This is required. An action entry field is located two to four spaces to the left of a selection field list choice. When an action entry field is to the left of a selection field list choice, the entry field can be one to eight positions in length.

The user may be given an option of relocating, and lengthening, the action entry fields to another portion of the line.

An option might also be to provide entry fields, under the column headings, in which a user can type an action code and parameters. This set of entry fields is in the panel body area containing instructions and column headings that do not scroll. There may be entry fields for several choice parameters when necessary to specify a choice.

Extendable Entry Fields. As an option, an application may make the action entry fields scrollable or extendable to allow a user to type action names that may extend beyond the normal length of a fixed-length entry field. An entry is extendable in the sense that a user can continue to type beyond the visible end of the field, typing over other information in the line.

An extendable action entry field is used instead of a scrollable entry field for typing an action and its parameters, if, for example, the entry field is long enough for the action but not its parameters.

When an extendable action entry field is used, the entry field delimiters are optional.

Text Cursor

The following factors apply:

- The cursor is initially located in the topmost and leftmost entry field in a list panel
- Immediately after a forward or backward action the cursor should be in the first entry field on the visible part of the panel, unless the previous cursor position is still visible.

- After a list has been processed and reappears, the cursor should appear by the last choice the user selected on the panel, unless that choice is no longer visible. In this case, the cursor should be in the first entry field in the panel.
- If the cursor was the top of the list of choices when a user requested enter, it should reappear at the top when processing is complete. For example, if an item was added to the beginning of the list, the list should be repositioned to start with the new item.
- When the application detects errors, the cursor should be in the first position of the first entry field in error.

12.7.3 How List Panels Work

A user interacts with a list panel in five ways:

1. A user selects choices by typing a slash character (/) in the action entry field and then switch to the action bar to request the pull-down containing the set of available actions. A user selects the action to work on all selected choices. A user may repeat this process to select a different set of choices and a different action to work on those choices. This method, required for all list panels, is compatible with menu panels that contain a multiple-choice selection field and it allows list panels to be used like menu panels for single actions.

2. A user types an action code, representing an action, in the action entry field next to each desired choice. Different actions may work on different choices. This is an application option.

3. An application should allow a user to type an equal sign (=) in the action entry field to indicate that the last action specified in the panel should also be applied to the choice alongside the equal sign. A user can repeat the equal sign on the panel. Each time it represents the last action or action code a user typed in an action entry field on the panel.

4. A list panel may contain entry fields for a user to type an action code or mnemonic and a choice not in the selection field. A user selects any available action for the choice typed in. A user may also request an action that requires input other than the action. This is an application option.

5. A user may also type commands and parameters in the action entry field if commands are supported.

Actions on list panels are not processed until a user requests ***enter***. Once entered, actions performed in the order displayed in the panel body.

Some actions, such as ***delete***, may require a user to confirm before processing starts. These actions may group together on a separate panel to display them for user confirmation. The user confirms the actions as a group but each action is performed in the order displayed on the panel.

As an application option, a user may interrupt list panel processing. If requested actions change the list of choices on the panel, the changes should appear the first time the panel is redisplayed by either the application or the user.

If the application detects processing errors, the action fields corresponding to the errors are highlighted with error emphasis when the list panel is redisplayed. The cursor is at the beginning of the first entry field in error.

When the application has processed all the actions, the list panel is displayed with the same view the user had of it when the action was requested. All changes resulting from the actions are shown to let a user know that the actions were processed and so the user can request other actions.

When the application has processed all the actions and the list panel is redisplayed, all previously processed single-character action entry fields and extendable action entry fields contain an asterisk (*) replacing the first character of the requested action. This shows the user what actions were processed and prevents actions from being resubmitted the next time the user requests enter.

The refresh action both updates the object list and restores defaults, if any exist, in the action entry fields.

12.8 Logo Panels

Used for display of company logo and any copyright notices. May be used as the panel from which a user signs on, enters user id and password. Password entry field should not display characters typed in.

12.9 Mixed Panels

There are times when it is necessary to create mixed panels (example: a panel body area from an entry panel with a panel body from a menu panel).

12.9.1 Creating Mixed Panels

When combining panel body areas to create mixed panels, the user should follow the procedures for each separate panel body area, and the panel elements they contain.

- Only one action bar, message area, command area, and function key area should be used.
- Nonscrollable panel body areas that are adjacent are both in the same panel body area, not separate panel body areas. This means a user can move between them using cursor-movement keys rather than a switch action.
- Each panel body area may have its own title and instructions.

Chapter 13

Color and Emphasis

This section covers default colors and emphasis. The approach described here is the basis from which more sophisticated use of color, shading, emphasis, etc. might be built.

13.1 Initial Display

When a panel is initially displayed, each panel element must appear in a particular color or intensity. As a user moves through a dialog with an application, the colors and emphasis may change to show the current state of an element.

13.2 Display Modes

The following display modes are discussed:

- Color display terminals (white, black, blue, and cyan palettes).
- Monochrome display terminals (white and black palettes).

For monochrome terminals, the black palette is a dark background that displays light text; the white palette uses a light background with dark text.

13.3 Color and Emphasis Tables

The following tables are recommended default colors and emphasis use. For color displays, the user should be allowed to switch between palettes.

The nonprogrammable terminals referred to in the tables are IBM-compatible 319X and 327X devices.

13.4 When to Use What Palette

The use of default palettes is guided in these ways:

- Panels displayed in primary and secondary windows, except help panels, use the ***white*** palette.

- Secondary windows that display help panels use the ***blue*** palette.

- Panels displayed in pop-up windows use ***cyan*** and ***white*** palettes, alternating at each level.
- Action bar pull-downs use the same foreground and background colors as the action bar.

- If a color display does not support backgrounds other than black, use the ***black*** palette. On nonprogrammable terminals, use the ***black*** palette for all windows.

13.5 Color and Emphasis Guidelines

Applications will at times need to vary from the color and emphasis guidelines stated here. The intent to use color and emphasis to make the application easier to use should be considered when deviating from these guidelines.

It is usually best to keep the number of different colors to a minimum.

13.6 Colors for Primary and Secondary Windows (Except for Help)

Programmable Terminals

White Palette

Panel Element	Color
Action Bar and Pull-down	
Background	Cyan
Choices	Black
Mnemonics	Underscore (note #1)
Panel Body	
Background	White
Protected Information	Blue
Brackets	Blue
Column Headings	Blue
Emphasized Text	Brown (note #5)
Entry Field box	Blue
Field Prompts	Blue
Group Headings	Blue
Instructions	Blue
Normal Text	Blue
Panel Titles	Blue
Panel ID	Blue
Choices	Black
Unavailable Choices	Reduced contrast
Mnemonics	Underscore (note #1)
Entry Field content	
Normal input	Black
Emphasized input	Brown
Scroll arrows and information	Blue
Separator line	Blue
Pop-up window border	Blue
Function Key Area	
Background	White (note #2)
Choices	Black

Colors for Primary and Secondary Windows, continued

Programmable Terminals

White Palette

Panel Element	**Color**
Selection Emphasis Section Cursor On choice Entry Field Selected Emphasis Selected/Cursored	 Black Text cursor in field Reverse color (note #3) Reverse color & outline box (note #3)
Error Emphasis	Red
Notification Messages Background Foreground	 White Black
Warning Messages Background Foreground	 Yellow Black
Critical Messages Background Foreground	 Red White

13.7 Alternate Palette for Windows

Programmable Terminals

Black Palette

Panel Element	Color
Action Bar and Pull-down	
Background	White
Choices	Black
Mnemonics	Underscore (note #1)
Panel Body	
Background	Black
Protected Information	Cyan
Brackets	Cyan
Column Headings	Cyan
Emphasized Text	Yellow (note #5)
Entry Field box	Cyan
Field Prompts	Cyan
Group Headings	Cyan
Instructions	Cyan
Normal Text	Cyan
Panel Titles	Cyan
Panel ID	Cyan
Choices	White
Unavailable Choices	Reduced contrast
Mnemonics	Underscore (note #1)
Entry Field content	
Normal input	White
Emphasized input	Yellow
Scroll arrows and information	Cyan
Separator line	Cyan
Pop-up window border	Cyan
Function Key Area	Black (note #2)
Background	White
Choices	

Alternate Palette for Windows, continued
Programmable Terminals

Black Palette

Panel Element	Color
Selection Emphasis Selection Cursor On choice Entry Field Selected Emphasis Selected/Cursored	 White Text cursor in field Reverse color (note #3) Reverse color and outline box (note #3)
Error Emphasis	Red
Notification Messages Background Foreground	 White Black
Warning Messages Background Foreground	 Yellow Black
Critical Messages Background Foreground	 Red White

13.8 Colors for Windows Displaying Help Panels

Programmable Terminals

Blue Palette

Panel Element	Color
Action Bar and Pull-down	
Background	Cyan
Choices	Black
Mnemonics	Underscore (note #1)
Panel Body	
Background	Blue
Protected Information	Cyan
Brackets	Cyan
Column Headings	Cyan
Emphasized Text	Yellow (note #5)
Entry Field box	Cyan
Field Prompts	Cyan
Group Headings	Cyan
Instructions	Cyan
Normal Text	Cyan
Panel Titles	Cyan
Panel ID	Cyan
Choices	White
Unavailable Choices	Reduced contrast
Mnemonics	Underscore (note #1)
Entry Field content	
Normal input	White
Emphasized input	Yellow
Scroll arrows and information	Cyan
Separator line	Cyan
Pop-up window border	Cyan
Function Key Area	
Background	Blue (note #2)
Choices	White

Colors for Windows Displaying Help Panels, continued

Programmable Terminals

Blue Palette

Panel Element	Color
Selection Emphasis	
Selection Cursor	
On choice	Black
Entry Field	Text cursor in field
Selected Emphasis	Reverse color (note #3)
Selected/Cursored	Reverse color and outline box (note #3)
Error Emphasis	Red
Notification Messages	
Background	White
Foreground	Black
Warning Messages	
Background	Yellow
Foreground	Black
Critical Messages	
Background	Red
Foreground	White

13.9 Colors for First Level Pop-up Windows

Programmable Terminals

Cyan Palette

Panel Element	Color
Action Bar and Pull-down	
Background	White
Choices	Black
Mnemonics	Underscore (note #1)
Panel Body	
Background	Cyan
Protected Information	Blue
Brackets	Blue
Column Headings	Blue
Emphasized Text	Yellow (note #5)
Entry Field box	Blue
Field Prompts	Blue
Group Headings	Blue
Instructions	Blue
Normal Text	Blue
Panel Titles	Blue
Panel ID	Blue
Choices	Black
Unavailable Choices	Reduced contrast
Mnemonics	Underscore (note #1)
Entry Field content	
Normal input	Black
Emphasized input	Yellow
Scroll arrows and information	Blue
Separator line	Blue
Pop-up window border	Blue
Function Key Area	
Background	Cyan (note #2)
Choices	Black

Colors for First-Level Pop-up Windows, continued

Programmable Terminals

Cyan Palette

Panel Element	**Color**
Selection Emphasis Selection Cursor On choice Entry Field Selected Emphasis Selected/Cursored	 Black Text cursor in field Reverse color (note #3) Reverse color and outline box (note #3)
Error Emphasis	Red
Notification Messages Background Foreground	 White Black
Warning Messages Background Foreground	 Yellow Black
Critical Messages Background Foreground	 Red White

13.10 Colors for Second Level Pop-up Windows

Programmable Terminals

White Palette

Panel Element	**Color**
Action Bar and Pull-down	
Background	White (note #4)
Choices	Black
Mnemonics	Underscore (note #1)
Panel Body	
Background	White
Protected Information	Black
Brackets	Black
Column Headings	Black
Emphasized Text	Black
Entry Field box	Black
Field Prompts	Black
Group Headings	Black
Instructions	Black
Normal Text	Black
Panel Titles	Black
Panel ID	Black
Choices	Black
Unavailable Choices	Reduced contrast
Mnemonics	Underscore (note #1)
Entry Field content	
Normal input	Black
Emphasized input	Black
Scroll arrows and information	Black
Separator line	Black
Pop-up window border	Black
Function Key Area	
Background	White (note #2)
Choices	Black

Colors for Second-Level Pop-up Windows, continued
Programmable Terminals

White Palette

Panel Element	**Color**
Selection Emphasis	
Selection Cursor	
On choice	Black
Entry Field	Text cursor in field
Selected Emphasis	Reverse color (note #3)
Selected/Cursored	Reverse color and outline box (note #3)
Error Emphasis	Black
Notification Messages	
Background	White
Foreground	Black
Warning Messages	
Background	White
Foreground	Black
Critical Messages	
Background	Black
Foreground	White

13.11 Colors for Nonprogrammable Color Terminals

Panel Element	Color
Action Bar and Pull-down	
Background	Black
Choices	White
Mnemonics	Underscore (note #1)
Panel Body	
Background	Black
Protected Information	
Brackets	Turquoise
Column Headings	Turquoise
Emphasized Text	Yellow
Field Prompts	Turquoise
Group Headings	Turquoise
Instructions	Turquoise
Normal Text	Turquoise
Panel Titles	Yellow
Panel ID	Blue
Choices	White
Unavailable Choices	Blue
Mnemonics	Underscore (note #1)
Entry Field content	
Normal input	Green
Emphasized input	Yellow
Scrolling arrows and information	Blue
Separator line	Turquoise
Pop-up window border	Turquoise
Function Key Area	
Background	Black
Choices	Blue
Selection Emphasis	
Selection Cursor	
Choices	Text cursor
Entry Field	Text cursor
Selected Emphasis	Yellow
Selected/Cursored	Yellow/text cursor
Error Emphasis	Red

Colors for Nonprogrammable Color Terminals, continued

Panel Element	Color
Notification Messages	
Background	Black
Foreground	White
Warning Messages	
Background	Black
Foreground	Yellow
Critical Messages	
Background	Black
Foreground	Red

13.12 Colors for Nonprogrammable Monochrome Terminals

Panel Element	Color
Action Bar and Pull-down	
Background	Black
Choices	White
Mnemonics	Underscore (note #1)
Panel Body	
Background	Black
Protected Information	
Brackets	Normal intensity
Column Headings	High intensity
Emphasized Text	High intensity
Field Prompts	Normal intensity
Group Headings	Normal intensity
Instructions	Normal intensity
Normal Text	Normal intensity
Panel Titles	High intensity
Panel ID	Normal intensity
Choices	Normal intensity
Unavailable Choices	Asterisk character
Mnemonics	Underscore (note #1)
Entry Field content	
Normal input	Normal intensity
Emphasized input	High intensity
Scrolling arrows and information	Normal intensity
Separator line	Normal intensity
Pop-up window border	Normal intensity
Function Key Area	
Background	Black
Choices	Normal intensity
Selection Emphasis	
Selection Cursor	
Choices	Text cursor
Entry Field	Text cursor in field
Selected Emphasis	Reverse color (note #7)
Selected/Cursored	Reverse color (note #7)
Error Emphasis	Reverse color (note #6)

Colors for Nonprogrammable Monochrome Terminals, continued

Panel Element	Color
Notification Messages Background Foreground	 Black High intensity
Warning Messages Background Foreground	 Black High intensity
Critical Messages Background Foreground	 Reverse color (note #6) Reverse color (note #6)

NOTES:

1. If underscore is not available for mnemonics, the color or shade is changed. If this can't be done and the mnemonic is not the first character of the choice, then a suffix should be added to the choice consisting of a mnemonic character in parenthesis.

2. When the panel body cannot be scrolled, the function key area and the panel body should be made the same color. When the panel body can be scrolled and there is a color boundary separator between the panel body and the function key area, the background and the choice colors of the function key area should be made the same as those specified for the action bar.

3. Selected emphasis for action bar choices on IBM compatible PCs may be vertical bars on each side of the selected choice.

4. A black separator line, or box, should be used to separate the action bar from the panel body when they both have the same background color.

5. A shade of yellow or brown--whatever stands out best on the screen-should be used.

6. Highlighted text should be used if reverse color is not available.

7. For screens that do not support reverse color, the only way to distinguish selected from unselected choices is by using selection indicators.

Chapter 14

Dialog Actions

Dialog actions are actions that have common meaning in all applications. These actions are assigned to keys so that a user can request them by pressing the assigned keys (or pointing to the key assignments in the function key area). Some of the actions can also be requested from an action bar pull-down or by typing a command in a command area.

The dialog actions are:

Cancel, Command, Enter, Exit, Prompt, Refresh, Retrieve, Display Panel IDs, Display Keys, Help.

4.1 Cancel

Cancel allows a user to back up in a dialog, one panel at a time, or to back up from a pull-down to the action bar.

- ***Cancel*** must be included in all pull-downs.
- ***Cancel*** should be included on panels inside pop-up windows, with the exception of information panels that contain **enter to continue** type instructions.
- ***Cancel*** may be included in other panels as an application option. It is recommended that ***cancel*** be included in all panels that are part of a sequence of panels which comprise a single unit of work.

14.1.1 Responses to Cancel

When a user cancels a pop-up window, the pop-up is removed and control returns to the previous panel, window, or action bar. Information entered by the user is discarded.

Repeated ***cancel*** requests will take a user back to the highest-level panel. At that point, another ***cancel*** has the same effect as the exit action.

If an application determines that significant information will be lost because of a ***cancel*** action, a confirmation message may appear, prompting the user to save or discard the information.

14.2 Command

The command dialog action allows a user to interact with a command area. See "The Command Dialog Action."

14.2.1 Enter

When a user finishes interacting with a panel that contains entry fields or selection fields, it has to be submitted to the application with a specific action request. The ***enter*** action tells the application to process the panel. The ***enter*** action is initiated by pressing the ***enter*** key or selecting ***enter*** in the function key area with a mouse. The ***enter*** action will always result in a visible response.

The ***enter*** action sends information from an action bar pull-down or from an entire panel to the application for processing. If the cursor is in the pull-down, the action selected by users in the pull-down determines how the dialog will continue. If the cursor is in the panel body, ***enter*** submits the entire panel. If the panel uses an implied action, the dialog continues. When no action is implied, the application asks a user to complete the panel by specifying an action.

14.2.1.1 Responses to Enter

The responses to the ***enter*** action vary from situation to situation. It is important that the panel title or instructions should identify any implied actions.

In cases where multiple actions are possible, it may be appropriate that an ***enter*** will cause the cursor to move to the action bar and cause a pull-down to appear. The user would then choose the action to be taken.

14.2.2 Exit

An application may provide a hierarchy of function levels. If so, it may provide several exit levels to allow a user to return to a particular function. ***Exit*** terminates the current function and returns a user to a higher level function. Repeated ***exit*** requests return the dialog to the highest level in the panel hierarchy that a user is aware of.

14.2.3 Prompt

When a user requests the ***prompt***, a pop-up window appears with information to help a user complete an entry field. See 17.0, "Prompt."

14.2.4 Refresh

When a user requests ***refresh***, the content of the current panel is restored to its original state or it is refreshed to reflect the current status (or a combination of both).

14.2.5 Retrieve

A user may request ***retrieve*** while a command area is visible to redisplay the last command that was issued.

14.2.6 Display Panel IDs

When a user requests the ***Display Panel IDs*** action, the panel IDs are turned on or off.

A user is allowed to turn the display of IDs on and off several ways, such as by selecting a pull-down choice, by entering a command, or by pressing a key.

14.2.7 Display Keys

When a user requests the display keys action, the function key area and the area at the bottom of a pull-down that contains key assignments change form. The default long-form changes to short-form (see 9.0 "Three Forms of Function Key Area"). The next display action request changes to a no display mode, the third display action request returns the function key area to the long-form.

14.2.8 Help

When a user requests Help, a help panel is displayed to assist the user to complete the dialog.

- *Extended Help*. When requested, a help panel appears containing information about the current task rather than help about a particular item on the panel. The extended help action appears only in help panels and pull-downs for the action bar help choice.

- *Help Contents*. When requested, a table of contents appears, organized by topic, of all the help information in the application. The help contents action appears only in help panels and pull-downs for the action bar help choice.

- *Help Index*. When requested, an index of the help information appears. The help index action appears only in help panels and pull-downs for the action bar help choice.

- *Keys Help*. When requested, a list appears containing the names and functions of all the keys used by the application. The keys help action appears only in help panels and pull-downs for the action bar help choice.

- *Tutorial Help*. If the application provides a tutorial, a user can access it by requesting the tutorial help action from the help pull-down or from help panels.

Chapter 15

Pop-up Windows

A ***pop-up window*** is a portion of the screen, enclosed by a border, in which a panel is displayed that extends the user's dialog with another window. Pop-up windows are associated with underlying windows and appear when the application wants to extend the dialog in the underlying window. A user must finish interacting with the pop-up window, unless it contains a help panel, before continuing with the dialog.

Pop-up windows have nothing to do with whether an application is used in a ***windowing environment*** or ***non-windowing environment***. Pop-up windows are used in both environments. Applications must use pop-up windows regardless of the environment.

Pop-up windows are, to a large degree, controlled by the application and not the user. A user cannot move or resize pop-ups.

An application may have a series of overlapping pop-up windows associated with an underlying window.

15.1 Pop-up Window Location

A pop-up window is attached to an underlying window and moves with the underlying window. It is possible to extend a pop-up beyond the boundary of the underlying window to which it is attached.

15.2 Initial Location of a Pop-up Window by Item-Adjacent Positioning

When a pop-up window is related to an item in the underlying window, position the pop-up adjacent to a related item, either below, above, to the right of, or to the left of the related item.

If the application scrolls the information in the underlying window to keep it visible when a pop-up window is displayed, the application must restore the information to its original position in the underlying window when the pop-up window disappears.

15.3 Initial Location of a Second Pop-up Window by Offset Positioning

When one pop-up window is displayed on top of another, vertically offset the pop-up window below the title of the underlying window so the title is visible. Horizontally offset the pop-up window two character spaces to the right of the beginning of the choices in the underlying window so the user can see which choice was selected.

15.4 Pop-up Window Layout

A double-line border should be placed around pop-up windows (if available, otherwise use single-line).

Pop-up windows are fixed in size and position relative to the underlying panel or other underlying pop-up window.

Pop-up windows should be kept small but large enough to display the information they must contain. Some blank space should be left inside the window border to set off the pop-up from the information in the underlying panel. Scrolling should be used if large amounts of information must be displayed.

15.5 Pop-up Window Content

Any of the defined panel types, except logo, may be used in a pop-up window. Pop-up windows can contain any panel element available (action bars, function key areas, entry fields). A window title bar will not be used in pop-up windows.

Chapter 16

Help

The purpose of ***help*** is to assist a user of an application do the job more efficiently. Help can remind a user about syntax or procedures for using a panel, or to explain a message.

Help is not meant to be a tutor. Tutorials teach a new user how to use an application. help assists a user recall how to use an application.

When a user requests help for a specific item, the help should be about that item. A user should also be able to request an index of help topics, a listing of keys and their functions, extended information about the panel being used, and general information about the help facility.

16.1 Help Location

If secondary windows are available, they should be used to display help panels, otherwise use pop-up windows.

The application determines the initial size of the help window.

Though not recommended, full-screen help panels can also be used.

16.2 Help Layout

Help panels can be of different styles (information panels, menu panels, list panels, or entry panels). See the guidance for these styles to determine layout.

16.3 Help Content

A help panel to be displayed should contain assistance for the item upon which the selection cursor is located. If there is no selection cursor, a general help panel must appear to explain what is in the panel.

Help panels should contain the normal panel elements and information for the panel type used, plus the following dialog actions.

Help provides:

- Contextual help about a specific field, an application panel, or about the help facility.
- Field help when the cursor is on a choice or entry field in an application panel or another help panel.
- Information about the application task, called *extended help*, when the cursor is not on an interactive selection or entry field.
- Information about the help facility when it is requested from a help panel and the cursor is not on an interactive selection or entry field.

Extended help. This produces a help panel containing information about the application panel from which users requested help.

Keys help. This produces a panel describing each key used by an application and its function. The listing does not have to indicate which keys are valid in the current state of the application. Large lists may be scrollable.

Help index. This produces an alphabetic index of all the help information in the application. When a user selects a help topic from the index, a help panel containing information about that topic appears.

Help contents. This option produces a table of contents of help information, organized by topic. The choice "Help Contents" should appear in the help action bar pull-down. A function key may be assigned to help contents.

Tutorial help. This option provides access to a tutorial that a user can get from the current panel.

All help information appears in the same window. When a user requests help, help index, extended help, keys help, or tutorial help, that information replaces the help currently in the help window.

16.4 How a User Requests Help

A keyboard user requests help by pressing the ***help*** key.

A mouse user may request help by pointing at ***F1=help*** in the function key area, and releasing the select button while the cursor is on a selection field choice or entry field.

Both keyboard and mouse users can request other forms of help by selecting the desired choice from the action bar pull-down for help.

16.5 How a User Interacts with Help Panels

The help panel assists a user with the item indicated by selection cursor. If the selection cursor is not on a choice or entry field, or if there is no selection cursor in the panel, the extended help panel that appears contains general information about the task or procedure the user is performing in the application panel.

If a user requests help while a help panel is displayed, the next panel provides information about an entry field or selection-field choice the selection cursor is on in the first help panel. If the first help panel contains no entry or selection field, or if the selection cursor is not in an interactive field, a panel appears telling a user how to use the help facility.

If the first help panel does contain an entry field or selection field, the panel must also support the extended help action. If a user requests this action, a help panel appears explaining the application panel from which the user originally requested help. This is the same panel a user sees if the first help request was from an application panel that had no entry or selection field or in which the cursor was not on an interactive field.

Regarding interaction with help in windows, the degree to which a user can interact with a help panel depends on whether the panel appears in a pop-up or secondary window.

16.6 Help in a Pop-up Window

A user interacts with a help panel in the same manner as with the panel type (information panel, menu panel, list panel, or entry panel) being used for the help panel. The help panel should be positioned so that the user can still interact with the field for which help was requested (user can switch back and forth between the help panel and the underlying field). This is the only exception where switching is allowed from pop-ups. The switch action should be assigned to a function key.

If the field in the underlying window is fully visible and not overlaid with the help pop-up window, the help pop-up remains displayed. Otherwise, it disappears. Upon a user request to switch back to the help panel, the help pop-up reappears.

A user's dialog with help panels ends and the pop-up window disappears when the user completes the field for which help was requested. Help also ends when a user cancels the first help panel or exits from any subsequent help panels. The underlying panel becomes interactive again.

16.7 Help in a Secondary Window

When secondary windows are available for help panels, a user interacts with them in the same way as with pop-up windows (see previous topic), with the following exceptions:

A user may move and resize secondary windows. They are initially positioned using the window-adjacent technique, as follows:

- If the primary window is full-screen, position the secondary window in a corner so that the secondary window covers up the least amount of information in the primary window

- If the primary window is not full-screen and both windows cannot be displayed without overlapping them, position the secondary window in a corner so that the least amount of overlap results.

- If both windows can be displayed concurrently without overlapping them, the preferred order for positioning the secondary window is:

  1. to the right of the primary window
  2. to the left of the primary window
  3. below the primary window
  4. above the primary window
  5. centered, beside the primary window, if space permits

- The interactive window is always the topmost window. When a user switches back to the window underlying the help window, the underlying window comes to the top and overlays the help window, partially or entirely covering the help window.

- A user can move and resize the two windows so that both the field needing help and the help information can be seen. Thus the user can switch back and forth between the two windows without causing an overlap.

16.8 Ending Help

A user can end help by requesting exit or by repeatedly requesting ***cancel*** to back out of any help panel.

16.9 Guidelines for Creating Help Panels

To increase usability and quality of help panels, the following guidelines should be followed:

- Help should be available at all times.
- Every panel in an application should have a help panel.
- Help panels should give immediate information about the current procedure or task a user wants to perform.
- Every help panel, except the extended help panel itself, should provide access to an associated extended help panel.
- If an application allows a user to enter commands, a help panel should appear with information about a particular command if a user types that command in the command area, leaves the cursor in the command area, and requests help. If a user does not type a command in the command area and requests help, a panel with general information about commands should appear.
- The title of help panels should contain the word ***help*** and identify the panel or items to which it applies.
- Dialog actions and function keys within the function key area should work the same way in help as they do in the rest of the application.
- When a user displays or removes help panels, the help facility must not interfere with the current dialog. Data entered by a user before requesting help must not be lost.

Chapter 17

Prompts

Prompt is a dialog action that assists a user in completing entry fields. It can save time for a user and reduce the chance of typing errors.

To use prompt, a user places the selection cursor on the entry field for which a list of possible entries is desired. When a prompt is requested, a pop-up window appears containing a menu panel with a single- or multiple-choice selection field.

When a user selects one or more choices from the selection field, the pop-up window disappears. The choice text is placed into the entry field as though a user had typed it there. Prompt gives a user the opportunity of recognizing and selecting the choice desired rather than having to remember all the choices and typing the one desired.

If a user selects more than one choice from a multiple-choice field, the application determines the sequence in which the choices are placed in the entry field.

The size of the pop-up window is an application design decision. The selection field in the pop-up window lists the possible entries for the entry field. The selection field may be scrollable.

A user can cancel the prompt pop-up window without selecting a choice. Canceling the pop-up window has no effect on the entry field.

If a user requests the prompt action when the cursor is not on an entry field, nothing happens. If the cursor is on an entry field but the field does not support the prompt function, a "beep" sounds. In addition, a message appears in a pop-up window or in the panel's message area. If the message appears in a message area, the message is removed on the next user action.

Prompt is not intended to be a substitute for the general-purpose directory and filelist capabilities of the operating system. It is intended to be used for application-specific field types.

An application option is to allow a user to request prompt when the cursor is in the command area. A list of available commands is displayed. A user can complete a command by selecting from the list. If command parameters are required, pop-up windows might be used to display permissible parameters.

Chapter 18

Messages

Messages are the means by which the computer communicates with users, delivering information provided by the application designer/developer.

18.1 Types of Messages

- ***Notification messages*** - provide feedback about the state of the system. No user respones is required. Should be used to notify a user that a process is executing (or has executed), significant changes in panel information, and periods of time when the user will have to wait.

- ***Warning messages*** - call a user's attention to a condition that may require action. User may respond with a Cancel or some other action

- ***Critical messages*** - specify a condition that requires users to act so the application can continue. User must respond with a specific action. Critical messages are used to tell a user that an imvalid action has been taken or that the system has detected an exception condition, requiring the user to take some action.

18.2 Message Location

It is recommended that messages be displayed in pop-up windows. The alternative is to display messages in a message area at the bottom of the panel body. A message area should be used when it is important that the message not overlay information in the panel. It is also appropriate to use the message area when a response from the user is not required.

Critical messages must be displayed in a pop-up window.

18.3 Message Layout

Panels created for messages must be information panels, menu panels, or entry panels. The message layout should follow the rules of the panel type being used (with the exception that panel titles are not used).

A message area can be any number of lines. Space must be allocated on a panel for the longest message.

18.4 Message Content

- Messages should not have a panel title.
- Information panels that are used to present notification messages do not contain the cancel action. The enter action is used to remove the panel.
- Notification messages that will be automatically removed will not contain a function key area.
- A "beep" should be used with warning and critical messages.

18.5 How a User Removes Messages

When possible, notification and warning messages are automatically removed when an action is detected. On nonprogrammable terminals, messages should disappear when a user presses a function key or requests ***enter***.

If automatic message removal is not possible, a user must remove the message with normal panel completion actions, with an entry panel, or some other appropriate means.

Warning messages are removed by normal panel interaction. For example, a warning message may tell a user that a field value is invalid, the user can remove the message by entering a valid value.

Critical messages can only be removed by user action that is not part of a normal panel interaction; they can not be removed automatically.

Messages that use an entry panel or menu panel must have the cancel action in the function key area.

18.6 Guidelines for Creating Messages

The following are guidelines for creating useful messages:

- Help should be provided for messages if more information might be needed.
- For messages that indicate an error, information should be provided to tell the user what is wrong and how the problem can be corrected. Error emphasis should be used to highlight the fields with errors.
- Messages should be short and concise and without jargon, abbreviations and acronyms. Messages may use mixed case.
- Messages should use consistent terminology.

- When a key name is used in a message, it should be spelled the same way it is printed on the key (e.g., ***Ctrl***, not ***Control***).

- Display a notification message to tell a user when an action has completed, unless there is some indication displayed for the user.

- When an action will take some time to complete, a notification should be displayed (e.g., ***Please wait***). Intermediate messages as an action becomes partially complete may also be appropriate.

- When asking a user to confirm an action, a *Yes* and *No* should not be given as choices; a user may be confused. Clearer choices would be:

  1. save and exit.
  2. exit without saving.

- Message numbers are optional.

18.7 Help for Messages

Help should be provided for messages just as it would be provided for other information. menu, or entry panels.

Help might refer to page in a user manual.

Chapter 19

Key Assignments

An application will make key assignments using the following guidelines and specifications. Key assignments for the IBM Enhanced Keyboard and System 370 keyboards will be covered.

19.1 Guidelines for Assigning Keys

Below are guidelines for assigning keys:

- Keys not covered here (unshifted keys, Shift+key, Ctrl+key, and Alt+key combinations) may be used by an application. The Alt, Ctrl, and Shift keys are used with other keys to change the meaning of those keys. The Alt, Ctrl, and Shift keys are not meant to be used individually.

- Keys already assigned by the operating system should not be used when making assignments.

- The keys listed in this section should not be reassigned or given duplicate key assignments.

- A user should be allowed to change function key assignments, as an application option. A user should be able to assign actions and parameters to any function key and to specify how the key will be labeled on the screen.

- If a function assigned to a key is common to several applications, then the function should be assigned to the same function key in all the applications.

- If a user presses a key that has no assigned function in the current panel, nothing happens unless otherwise specified.

19.2 Key Assignment Tables

The following tables indicate what function to assign to a specific key or key combinations.

Function	Key Name(s)	Keyboard Number
Backspace		
PC	←Backspace	15
System 370	←Backspace	15
Backtab		
PC	|←	44+16 or 57+16
System 370	|←	44+16 or 57+16
Backward (note #1)		
PC	Page Up or F7	85 or 118
System 370	F7	118
Beginning of Data		
PC	Ctrl+Home	56+80 or 64+80
System 370	N/A	N/A
Beginning of Field		
PC	Ctrl+←(left arrow)	58+79 or 64+79
System 370	N/A	N/A
Beginning of Line		
PC	Home	80
System 370	N/A	N/A
Cancel (note #4)		
PC	Esc or F12	110 or 123
System 370	F12	123
Command		
PC	Shift+F9	44+120 or 57+120
System 370	F21	44+120 or 57+120
Delete (character)		
PC	Delete	76
System 370	Delete	76
Display Keys		
PC	Shift + F1	44+112 or 57+112
System 370	F13	44+112 or 57+112

Function	Key Name(s)	Keyboard Number
Down Arrow	↓	84
PC	↓	84
System 370		
End of Data		
PC	Ctrl+End	58+81 or 64+81
System 370	N/A	N/A
End of Field		
PC	Ctrl+→(right arrow)	58+89 or 64+89
System	N/A	N/A
End of Line		
PC	End	81
System 370	N/A	N/A
Enter		
PC	Enter	108
System 370	Enter	64 or 108
Enter/New Line		
PC	↵Enter	43
System 370	N/A	N/A
Erase to End of Field		
PC	Ctrl+Delete	58+76 or 64+76
System 370	Erase EOF	81
Exit (note #2 & #4)		
PC		
Application level	Shift+F3	44+114 or 57+114
Function level	F3	114
System 370		
Application level	F15	
Function level	F3	44+114 or 57+114
		114
First Field on Screen		
PC	N/A	N/A
System 370	Home	80
Forward		
PC	Page Down or F8	86 or 119
System 370	F8	119

Function	Key Name(s)	Keyboard Number
Help (note #3 & #4)	F1	112
PC	F1	112
System 370		
Help Contents	Shift+F11	44+122 or 57+122
PC	F23	44+122 or 57+122
System 370		
Extended Help		
PC	F2	113
System 370	F2	113
Help Index		
PC	F11	122
System 370	F11	122
Keys Help		
PC	F9	120
System 370	F9	120
Insert		
PC	Insert	75
System 370	Insert	75
Left (note #1)		
PC	Ctrl+Page Up	58+85 or 64+85
System 370	F19	N/A
Left Arrow		
PC	←	79
System 370	←	79
New Line/Enter		
PC	↵Enter	43
System 370	N/A	N/A
New Line		
PC	N/A	N/A
System 370	↵New Line	43
Prompt		
PC	F4	115
System 370	F4	115

Function	Key Name(s)	Keyboard Number
Refresh		
PC	F5	116
System 370	F5	116
Retrieve		
PC	F9	120
System 370	F9	120
Right (note #1)		
PC	Ctrl+Page Down or Shift+F8	58+86 or 64+86 44+119 or 57+119
System 370	F20	44+119 or 57+119
Right Arrow		
PC	→	89
System 370	→	89
Scroll Lock		
PC	Scroll Lock	125
System 370	N/A	N/A
Switch Backward		
PC	Shift+F6	44+117 or 57+117
System	F18	44+117 or 57+117
Switch Forward		
PC	F6	117
System 370	F6	117
Switch to Action Bar		
PC	F10	121
System 370	F10	121
Tab		
PC	→|	16
System 370	→|	16
Up Arrow		
PC	↑	83
System 370	↑	83
Word Left		
PC	Alt+←(left arrow)	60+79 or 62+79
System 370	Alt+←(left arrow)	60+79 or 62+79

Function	Key Name(s)	Keyboard Number
Word Right		
PC	Alt+→(right arrow)	60+89 or 62+89
System 370	Alt+→(right arrow)	60+89 or 62+89

Key Assignments for PCs Emulating System 370 Keyboards

Function	Key Name(s)	Keyboard Number
Backspace	Backspace	15
Backtab	|←	44+16 or 57+16
Backward	F7	118
Beginning of Data	N/A	N/A
Beginning of Field	N/A	N/A
Beginning of Line	N/A	N/A
Cancel	F12	123
Command	(Shift)+F9	44+120 or 57+120
Delete (character)	Delete	76
Display Keys	(Shift)+F1	44+112 or 57+120
Down Arrow	↓	84
End of Data	N/A	N/A
End of Field	N/A	N/A
End of Line	N/A	N/A
Enter	Ctrl	64
Erase to End of Field	End	81
Exit Application level Function level	 (Shift)+F3 F3	 44+114 or 57+114 114
First Field on Screen	Home	80
Forward	F8	119

Help Help Contents Extended Help Help Index Keys Help	F1 (Shift)+F11 F2 F11 F9	112 44+122 or 57+122 113 122 120
Insert	Insert	75
Left	(Shift)+F7	44+118 or 57+118
Left Arrow	← (left arrow)	79
New Line	↵Enter	43
Prompt	F4	115
Refresh	F5	116
Retrieve	F9	120
Right	(Shift)+F8	44+119 or 57+119
Right Arrow	→	89
Scroll Lock	N/A	N/A
Switch Backward	(Shift)+F6	44+117 or 57+117
Switch Forward	F6	117
Switch to Action Bar	F10	121
Tab	→|Tab	16
Up Arrow	↑	83
Word Left	Alt+←(left arrow)	60+79 or 62+79
Word Right	Alt+→(right arrow)	60+89 or 62+89

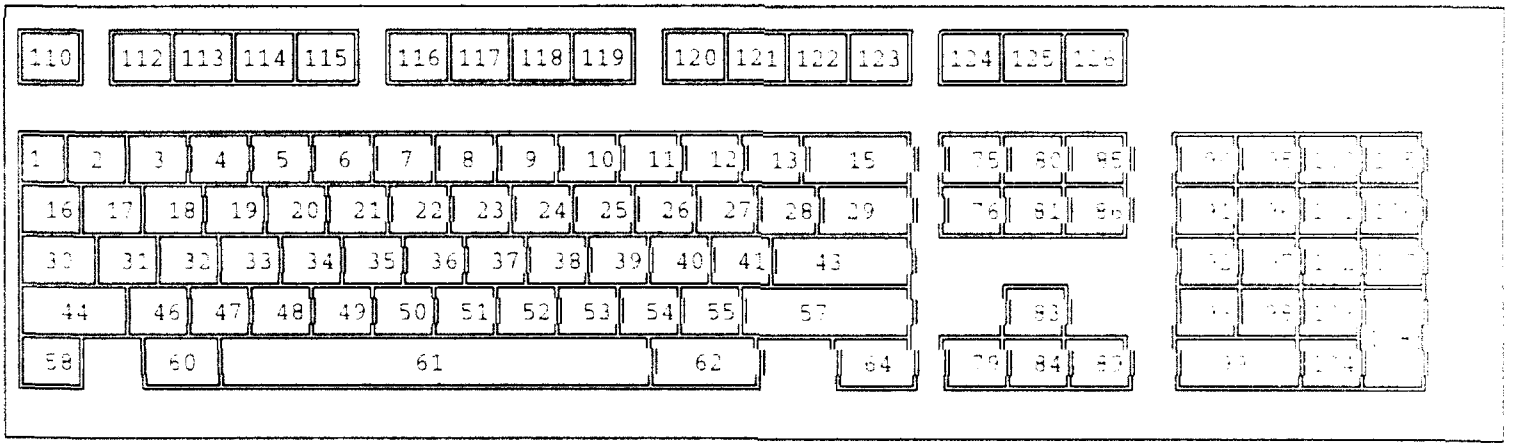
NOTES:

1. On some systems, two key combinations are assigned to this function. The default is that F7 is the same as Page Up and F8 is the same as Page Down. It is an application option to allow a user to assign separate functions to the F7 and F8 keys.

2. ***Application level*** and ***Function level*** refer to multiple exit points in the dialog hierarchy. ***Function level*** is the next level of exit as a user steps back through the application function by function. ***Application level*** is the farthest point to which a user can return, the starting point for the application. In effect, F15 is a fast-path exit to the highest level, bypassing the intermediate exit points.

3. The keys defined for help panels may be assigned to other functions in panels not used for help.

4. These keys are reserved, and cannot be reassigned to other functions.



Layout of the IBM Enhanced Keyboard With Keyboard Numbers